AMIGA

IL MENSILE JACKSON PER GLI UTENTI DI AMIGA

INSERTO LE PAGINE DI Amiga Transactor EDIZIONE ITALIANA

- Virus Protection Toolbox
- Superbase Professional 3.0
- Pen Pal
- Il Bridgeboard
- Dal 6502 al 68000
- Viaggio esplorativo alle 8520
- AmiExpo
- Expoedit

Spedizione in abbonamento postale Gruppo III/70 Taxe perçue (Tassa riscossa) Milano Ferr. Corr.

AMIGA MAGAZINE
ANNO 3 - NUMERO 15
SETTEMBRE 1990

Sommario

ECCO I FORTUNATI VINCITORI DEI FAVOLOSI

DEL CONCORSO ABBONAMENTI JACKSON 89•90

1° PREMIO (Sei giorni alle Mauritius, viaggio e soggiorno per due persone a La Pointe aux Canonniers organizzato e offerto dal Club Med) è stato vinto da: 3M ITALIA SpA - Serv. Informazione - 17016 FERRANIA (SV) • **2° PREMIO (una settimana ai Caraibi, viaggio e soggiorno per due persone a La Caravèlle organizzato e offerto dal Club Med) è stato vinto da:** QUAGLIERI SANTE - Via Dei Gracchi 285 - 00192 - ROMA **3° PREMIO (una settimana in Turchia, viaggio e soggiorno per due persone a Kemer organizzato e offerto dal Club Med) è stato vinto da:** PELLEGRINI SILVANO - Porta Catena 10 - 44100 FERRARA •

100 raccoglitori per floppy disk da 3" 1/2 (offerti dalla MEE S.p.A.) sono stati vinti da: IPSIA "B.CELLINI" - Via Masaccio 8 50136 FIRENZE • COLARUSSO ANGELO - Via Peircelli 159 - 82028 S.BARTOLOMEO IN GALDO (BN) • ANNOSCIA NATALBERTO c/o CABLER srl - Via Fermi 35 - 20083 GAGGIANO (MI) • I.T.C. "VITT.EMANUELE II" di GENOVA - L.go Zecca 4 - 16124 GENOVA • I.T.C. "A. MONTI" - Via S. Lo Rizzo 71 - 00128 ROMA • LINOTTO - SELTERING SpA - Via Cipro 33 - 25125 BRESCIA • CHIROLI ELIO - Off. Mecc. Chiroli Virginio - Via Genova 2 - 26015 SORESINA (CR) • TORRE VITTORIO - Via Oropa 133 - 10153 TORINO • TORRICELLA STEFANO - Via Palladio 10 - 30027 S.DONA' DI P. (VE) • TORCIANO PIERO - Via Nuoro 42 - 10137 TORINO • TOMMASETTI ALFONSO -Via Trento P. dei Pini 34 - 84016 PAGANI (SA) • TOMIELLO MAURIZIO - Via Ca da Po' 11 -15050 S.GIULIANO N. (AL) • CABIATI ERNESTO - Via Puggia 31A/20 - 16131 GENOVA • CARBONE LUCIANO - Piazza Colombo 2A/19 - 16121 GENOVA • LEASINT LEASING INTERN. SpA- Corso Monforte 19-20122 MILANO • DEL TORRE ALDO - Via Colugna 97/7 - 33100 UDINE • OLDANI MAURIZIO - Via Castelletto 23 - 20012 CUGGIONO (MI) • OLIVA DAVIDE - Via Sergio Bovo 5 - 28027 ORNAVASSO (NO) • CONTEMORI PIERLUIGI - Via A.Manzoni 27-50018 SCANDICCI (FI) • TORELLI STEFANO - Via Zanchi 3A - 24040 STEZZANO (BG) • CONTI GRAZIA - Via Pasubio 32 - 20081 ABBIATEGRASSO (MI) • PALETTA MARCO - Via dei Coronari 101 - 00186 ROMA • PALLAVICINI LUCA - Via Marco Polo 11 - 37138 VERONA • FRANZE' ANTONIO - Via C. Amatruda 18 - 88074 CROTONE (CZ) • CAPPELLI STEFANO - Via Dei Mulini 55 - 47100 FORLI' • PALMERA SpA - Via Palleri 3 - 16125 GENOVA • FRAQUELLI FULVIO - Via Monte Scale 1 - 23030 SEMOGO VALDIDENTRO (SO) • PALMA FRANCESCO - Via A. Volta 9 - 67051 AVEZZANO (AQ) • TORNIELLI ANGELO - Via Salvator Pianeli 63 - 20162 MILANO • FRANZONI ANDREA -Via Casa Buccelli 2 - 41040 S.MICHELE (MO) • CONTI PAOLO - Via Lattanzio 88 - 00136 ROMA • OLIVERO LUCA - Via Giardinasso 13 - 13040 SALUGGIA (VC) • COMPAGNIN PIETRO - Via D.Alighieri 12-35040 VESCOVANA (PD) • BENETTI PIERLUIGI - Via Don Facchetti 5 - 25085 GAVARDO (BS) • MORINI GLAUCO c/o BERCHI srl - Via Provinciale 32/a - 43038 SALA BAGANZA (PR) • DOLOBATI MAURIZIO - Via C. Mazza 20 - 44100 FERRARA • DREON LUIGI - Via G.Stucky 7 - 30025 FOSSALTA PORTOGR. (VE) • DONES DARIO - Via Clerici 12 - 20032 CORMANO (MI) • BRIOLINI GIOVANNI - Via del Picchio 18 - 40141 BOLOGNA • FERRARI GIORGIO c/o ULTRAFLEX srl - Via Sturla 32 - 16131 GENOVA • GIROLOMONI LUIGI - Via Cerbara 40 - 61030 MONTEMAGGIORE (PS) • BRASCA LEO - Via V. Di Marco 10 - 87070 VILLAPIANA SCALO (CS) • SCUDERI FRANCESCO - Via I. Nievo 11 - 33100 UDINE • SCUOLA MEDIA "C.Gouthier" - 10063 PEROSA ARGENTINA (TO) • SCUOLA MEDIA STATALE - Via Al Donatore di Sangue 1 - 20010 MARCALLO CON CASONE (MI) • DONVITO PASQUALE - V.le Del Tramonto 5/A - 33078 S.VITO AL TAGLIAMENTO (PN) • DOSIA ALDO FRANCO - Via Venezia 16 - 03030 PIEDIMONTE S.G.(FR) • CICCALE' MARCELLO-Via B.R.Margherita 44-63010 M.S.PIETRANGELI (AP) • COMATIC SpA- Via L.Da Vinci 21-Loc.Bodutto-25064 GUSSAGO(BS) • ITCG P.BELMESSERI - Via Malaspina - 54027 PONTREMOLI (MS) • SACCHI ENRICO - P.zza A. Moro 14 - 60020 AGUGLIANO (AN) • MAGRO ANTONELLO - Via Alpe Devero 23 - 28037 DOMODOSSOLA (NO) • CHILLON CARLO - Via D.Alighieri 12/A - 35010 VILLAFRANCA (PD) • CHIODINI ALESSANDRO - Via S.Pellico 32 - 20010 VITTUONE (MI) • MAESTRI ANGELO MAURIZIO- Via Fra' Cristoforo 2-20142 MILANO • CAPPONI G.PAOLO - Via Riva d.Garda 98 int.6 - 41100 MODENA • BRAVI SETTIMIO - Via Apecchiese 4 - 61048 S.ANGELO IN VADO (PS) • TAGLIAZUCCHI PIERLUIGI - Via F.Nova 34 - 20159 MILANO • TANA MICHELE - Via A. Noli 6 - 24100 BERGAMO • TAMBURELLO VINCENZO - Via F.Crispi 50-92017 SAMBUCA DI SICILIA (AG) • TANSI CARLO - Via I Maggio 10 - 20010 BAREGGIO (MI) • RAMIN ANTONIO - Via Vergerio 15/A - 35126 PADOVA • RASELLI FRANCO - Strada Fontanile 12 - 15048 VALENZA (AL) • REGGIANI ATHOS - Via S. Carlo 30 - 41049 SASSUOLO (MO) • I.T.I.S."G.CASO" - Via V. Caso - 81016 PIEDIMONTE MATESE (CE) • BERTOLINI GIUSEPPE - Via Magellano 4 - 24100 BERGAMO • ORSI P.LUIGI - Via Galvani 27 - 15100 ALESSANDRIA • ORTOLANO SANDRO - Via Manzoni 48 - 33032 BERTIOLO (UD) • SIMEONATO FABIO - Via Del Santo 2/A - 35010 S. GIORGIO D.PERTICHE (PD) • CASALBONI PIERO - Via Tommaso Gulli 241 - 48100 RAVENNA • FOPPOLI LUIGI - Via S. Andrea 27 - 25062 CONCESIO (BS) • FORCINA FILIPPO - Via d.Conca I Trav.18-04023 FORMIA (LT) • FORTUNATI OSVALDO - Via Lagoscuro 25 -19020 CEPARANA (SP) • BERTOLINO MARIO - Via Vinzole 7 - 28068 ROMENTINO (NO) • RANUCCI ERNESTO - Via E.Astuti 121 - 84014 NOCERA INF. (SA) • FORLENZA MARIO - Via Don Gonella 17-24052 AZZANO S.PAOLO (BG) • BARDELLE MAURO - Strada del Cascinotto 55 - 10156 TORINO • CANAGALLO FRANCO - C.so Lecce 24 - 10143 TORINO • CANALINI G.PAOLO - Via G. Deledda 2 - 41053 MARANELLO (MO) • DOTTI PIERFRANCO - Via di Villa Pepoli 20 - 00153 ROMA • FORESI ROMANO - Via Buonarroti 21 - 62010 MONTECOSARO SC. (MC) • CORONA GIANCARLO - Via Pisa 16 - 31100 TREVISO • COSTANZO ALBERTO - Via Livorno Ferraris 4- 13041 BIANZE' (VC) • BARONI ANGIOLO - Via Rosemberg 1 - 29010 CASTELVETRO P. (PC) • MILAZZO ROBERTO - V.le M.F.Nobiliare 16/22 - 00175 ROMA • MILANESI VALERIA - Via Bernina 43 - 20158 MILANO • LUMINA PIERANGELO - Via Correggio 12 - 24068 SERIATE (BG) • MARCHETTI CORRADO - Via Marco Polo 36 - 41100 MODENA • MARCHINI DANTE - P.le Giacopelli 5 - 43100 PARMA • VARUTTI SERGIO - Via S. Giacomo 180 - 33030 COSEANO (UD) • VENTURI FABRIZIO - Via Donacato 1 - 52036 PIEVE S.STEFANO (AR) • VENZA GIANCARLO - Via Citta' d.Pieve 19 - 00191 ROMA • VECCHIO COSIMO - Vico Francavilla 24 - 72024 ORIA (BR) • VENEZIANI P.GIORGIO - Via Valla 29 - 29100 PIACENZA • VENTURA FRANCESCO - Via Matteotti 4 - 92100 AGRIGENTO • BERTINI GRAZIANO - Via Pesciatina 54/A - 50054 PINETE FUCECCHIO (FI) • BERTOGLIATTI SERGIO - Via Mazzini 35 - 22040 BALLABIO (CO) • PARRAVICINI ALESSANDRO - Via Giovanni XXIII 5 - 20021 BOLLATE (MI) • VAVASSORI S.BATTISTA - Via Costituzione 13 -25038 ROVATO (BS) • PARRINI ROBERTO - Via U. Benzi 7 - 53100 SIENA •

100 raccoglitori per floppy disk da 5" 1/4 (offerti dalla MEE S.p.A). sono stati vinti da: PALUMBO M. ROSARIA - Via V. Mosca 11 - 80129 NAPOLI • CAMERA VINCENZO - Via Dante De Blasi 99 Int. 5B - 00151 ROMA • BENINI FERNANDO - Via Pagano 5 - 47037 RIMINI - BELLARIVA (FO) • VAZZOLER IVANO - Via Tempio Votivo 18 - 31010 PONTE D.PRIULA (TV) • MARCHETTO CLAUDIANO - Via Piave 48 - 37036 S.MARTINO B. ALBERG (VR) • FORMOSA GIOVANNI - Via Paolo di Dono 145 - 00143 ROMA • PASQUALOTTO FRANCO - Via Montello 13 - 31020 VIDOR (TV) • CASATI PAOLO - Via B.Da Montelupo 32/L - 50142 FIRENZE • ALFIERI MARCO - Via Mastri Muratori - 21040 GONATE OLONA (VA) • ANNARELLI SAURO - Via Carlo Meyer 18 - 57127 LIVORNO • PURPURA FAUSTO - Bivio Foresta 238 - S.S.113 - 90044 CARINI (PA) • MINARDI CLAUDIO - Via S.Lucia 18 - 48018 FAENZA (RA) • SACCO RENATO - Via Romera 10 - 15047 SPINETTA MARENGO (AL) • CARCANO LUIGI - Via P. Nenni 14 - 27011 BELGIOIOSO (PV) • FORMENTI ANDREA - Via Finiletto 19 - 46039 VILLIMPENTA (MN) • PERICOLI PAOLO - Via C.Dell'Acqua 41 - 20025 LEGNANO (MI) • GALASSI STEFANO - Via Sebenico 23 - 47041 BELLARIA (FO) • IPSIA " ERCOLE I D'ESTE"- Via Roversella 22- 44100 FERRARA • DONATINI VALTER -Via P.Caiani 21 - 50032 BORGO S.LORENZO (FI) • CARETTONI ROBERTO - Via Europa 21 - 20010 POGLIANO MIL. (MI) • ITIS "G.MARCONI" - Via IV Novembre 122 - 29100 PIACENZA • ERREOESSE SISTEMI di UMBERTO ROSANELLI - Via Cimabue 48 - 21100 VARESE • ERRICO MARIO - Via Sestese 77/3 - 50151 FIRENZE • PAVIA MAURO - Via Gobetti 3 - 10024 MONCALIERI (TO) • LICEO SCIENTIFICO STAT. "G.GALILEI" - 38051 B.VALSUGANA (TN) • LISI GIUSEPPE - Via F.lli Di Dio 182 - 20099 SESTO S.GIOV. (MI) • LOMBARDI BRUNO - Via Monte Tabor 63 - 21100 VARESE • LOMBARDI LEONARDO - L.go Impisi 10 - 71023 BOVINO (FG) • ESPOSITO ANTONINO - Via S.Cesareo 81- 80067 SORRENTO (NA) • VELLANI ALBERTO - Via Bonaccolsi 7 - 41012 CARPI (MO) • LOLLI P.LUIGI - Via D'Azeglio 853 - 41050 ROCCA MALAT. (MO) • IST. TECN. COMM. STAT. "L.EINAUDI" - Via Solferino 50 - 63018 PORTO S.ELPIDIO (AP) • CARAMELLA LUCA - Via Priv. Belvedere 2 - 28026 OMEGNA (NO) • DORATO LORENZO - Via d. Scuole 18 - 10020 BRUSASCO (TO) • LAMIANI AURELIO - Via Stoppani 25 - 20129 MILANO • LENOCI GIACOMO - Via E. Rivela 14 - 70021 ACQUAVIVA D. FONTI (BA) • VECCHIARELLI SANDRO - Via Bicocchi 1 - 58022 FOLLONICA (GR) • SCUOLA ELEMENTARE - Via C.Battisti 1 - 10030 RONDISSONE (TO) • ERO ELECTRONIC SpA - Via N.Bixio 47/49 - NOVATE MIL. (MI) • LANTERI ADRIANO - Via Peri 11 - 18030 POGGIO DI SANREMO (IM) • LAMPA MASSIMO - Via A.Regolo 1 - 80125 NAPOLI • LAZZARONI ELIO - V.le Romagna 29/D - 20092 CINISELLO B. (MI) • LATTUADA LUIGI - Via Piave 5 - 20051 LIMBIATE (MI) • LANDI ENRICO - Via Roma 52 - C. Post. 83 - 84091 BATTIPAGLIA (SA) • LO BUE GIANFRANCO - Via Michelangelo 5 - 92020 CASTROFILIPPO (AG) • GUALZETTI MASSIMO - Via Stelvio 49 - 23100 SONDRIO • FRATINI LUCIANO - Via L. Da Vinci 44 - 50018 SCANDICCI (FI) • FUMAGALLI UGO - Via Garibaldi 84 - 20055 RENATE B. (MI) • LANZA MASSIMO - Via Marconi 24 - 37053 CEREA (VR) • DE PRETTO - ESCHER WYSS srl - Via D. Manin 16/18 - 36015 SCHIO (VI) • DE RITIS GABRIELE - Via Cellaro 1 - 03039 SORA (FR) • DE ROSE FRANCESCO - Via Del Gabbiano 4 - 09043 MURAVERA (CA) • FILIPPI STEFANO-Via Dei Freli 8-31020 SERNAGLIA D/BATT.(TV) • BIGLIARDI CLAUDIO-Via Giovanni XXIII 6-20090 BUCCINASCO (MI) • CESARE BRUNO - Via Borgoratti 24/36 sc. D - 16132 GENOVA • CESTARO LUIGI - Via A. Costa 11 - 30038 SPINEA (VE) • LAURI GIOVANNI - I Trav. Licola Patria 65 - 80072 ARCO F. (NA) • LATORRE STEFANO - Via Tadino 3 - 20124 MILANO • LAVIA LUIGI - Via Romana Vecchia 54 -51010 CHISINA UZZ. (PT) • FORNASARI GIACOMO - Via Fratti 20 - 43100 PARMA • MANENTI LUCIANO c/o NEP srl - Via V. Monti 25 -20016 PERO (MI) • MANCUSO MAURO - Viale Sicilia 10 bis - 21052 BUSTO ARSIZIO (VA) • MANINI ETTORE - Via Novara 21 - 28039 VARZO (NO) • BONACORSI PAOLO - Via G.Di Vittorio 6 - 58100 GROSSETO • NOVARA EZIO - Via Don Murialdo 24 - 10142 TORINO • NOVELLI ROBERTO - Via Duchessa Jolanda 36 - 10138 TORINO • ESPOSITO GREGORIO - Via Del Progresso 314 - 88046 LAMEZIA T. (CZ) • GARLASCHELLI EMANUELA - V.le Lazio 7 - 20135 MILANO • MARRI PAOLO - L.go Mons. D'Ardia 12 - 00053 CIVITAVECCHIA (RM) • GENEROSO NICOLA - Via Puglie Coop. Alfa Romeo Is. 3 - 80038 POMIGLIANO D'ARCO (NA) • CADDEO STEFANO - Via Fontane 18 - 09010 SANTADI (CA) • GERSICH STEFANO - Via Diaz 15/8 - 30030 SALZANO (VE) • MANICARDI ARGIMORO - Via V. Veneto 45 - 41016 NOVI DI MODENA • PAVIA MAURO - Via Gobetti 3 - 10024 MONCALIERI (TO) • CAPRARA SIMONE - Via Ragazzi del 99 n.5/4 - 37134 VERONA • CAPUTO GIUSEPPE - Via Roma - 81040 FORMICOLA (CE) • CAPRIOGLIO FRANCO - Via Stevani 17 - 15030 S. MARTINI D/ROS. (AL) • LATINI MARIO - Via F. Stilicone 213 - 00175 ROMA • MENEGALDO GIOVANNI - Via Diaz 86/H - 38060 MATTARELLO (TN) • VOTO CARMINE - Via Italia 229 - 84098 PONTECAGNANO (SA) • GENINATTI MARIA - Via Borgomanero 14 - 10145 TORINO • GENTILI FABIO - Via Morosina 6/E - 30030 CAMPALTO (VE) • MENSI DANILO - P. V.Emanuele 17 - 15053 CASTELNUOVO S. (AL) • RUSSO Prof. FRANCESCO A. - Via Merliano 13 - 80035 NOLA (NA) • DESSALVO GIACOMO - Via Laviosa 5-5 - 16156 GENOVA - PEGLI • LAZZAROTTO LINO - Via St. M. De Corleans 40 - 11100 AOSTA • MORICONI MASSIMILIANO - Via C. Battisti 32 - 62032 CAMERINO (MC) • BRESCIANI M. GRAZIA - Via Cadore 31 - 13100 VERCELLI • BRESSAN A. MARIA - Via Villoresi 66 - 20022 CASTANO P. (MI) • MORINI FURIO - Via Olmi 120 - 50050 STABBIA (FI) • MOTTA G.ANGELO - Via Roma 28 - 27023 CASSOLNOVO (PV) • GALESSO ANDREA - Via L. Rizzo 76 - 35129 PADOVA • I.P.S.I.A. "MORETTO" - Via Apollonio 21 - 25124 BRESCIA • SACCHI MARCO A. - Viale Dante 9 - 22053 LECCO (CO) • SANTARELLI DOMENICO - Via Ariana 304 - 00031 ARTENA (RM) • ALBERTONI ENRICO - Via Ravizza 11 - 35136 PADOVA • ALOI ALVARO - Via Palermo 14 - 10097 REGINA MARGH. (TO) • BARDELLA MARCO - C.so Novara 4 - 10152 TORINO • VENDRAME CARLO - Via Busnello 12 - 31052 MASERADA (TV) • BAREA MANOLO - Via S. Pellico 18 - 34070 TURRIACO (GO)

Direttore Responsabile: Paolo Reina
Coordinamento Tecnico e Redazionale: Massimiliano Anticoli
Tel. 02/6948287
Redazione: Daniele Cassanelli (On Disk), Romano Tenca (Transactor)
Segreteria di redazione: Elena Ferré - Tel. 02/6948254
Art Director: Marcello Longhini
Copertina, Grafica e Impaginazione elettronica: Cristina Turra
Collaboratori: Lucio Bragagnolo, Cesare Palmieri, Liliana Simonetti, Stefano Riva, Sebastiano Vigna, Stefano Albarelli, Mauro Ballocchi, Aldo e Andrea Laus, Gianni Biagini, Renato Acciardi, Stefano Paganini, Marshal M. Rosenthal

Group Publisher: Pierantonio Palerma
Direzione Coordinamento Operativo: Graziella Falaguasta
Publisher Area Consumer: Filippo Canavese
Pubblicità: Filippo Canavese
Direzione Sviluppo Pubblicità: Walter Bussolera

SEDE LEGALE
Via P. Mascagni, 14 - 20122 Milano

DIREZIONE - REDAZIONE
Via Pola, 9 - 20124 Milano - Tel.: 02/69481
Fax: 02/6948238 Telex 316213 REINA I

PUBBLICITA'
Via Pola, 9 - 20124 Milano - Tel.: 02/6948254
ROMA - LAZIO E CENTRO SUD Via Lago di Tana, 16
00199 Roma
Tel.: 06/8380547 - Fax: 06/8380637

INTERNATIONAL MARKETING
Tel.: 02/6948233

DIREZIONE AMMINISTRATIVA
Via Rosellini, 12 - 20124 Milano Tel.: 02/69481
Fax: 02/6948238

UFFICIO ABBONAMENTI
Via Rosellini, 12 - 20124 Milano - Fax: 02/6948489
Telex 333436GEJ IT - Tel.: 02/6948490 (nei giorni di martedì, mercoledì, giovedì. 14.30 - 17.30)

Prezzo della rivista: L. 14.000 prezzo arretrato L. 28.000
Non saranno evase richieste dei numeri usciti prima del 1/1/89
Abbonamento annuo Italia L. 124.000, Estero L. 248.000
I versamenti vanno indirizzati a:
Gruppo Editoriale Jackson SpA
Via Rosellini, 12 - 20124 Milano, mediante l'emissione di assegno bancario o per contanti. L'abbonamento può essere sottoscritto anche utilizzando il c/c postale 11666203

CONSOCIATE ESTERE
GEJ Publishing Group Inc. Los Altos Hills
27910 Roble Blanco
94022 California - Tel.: (001 - 415 - 9492028)
Grupo Editorial Jackson - Calle Alcantara, 57
280016 Madrid - Tel.: 1/4017365

Stampa: F.B.M. (Gorgonzola)
Fotolito: Foligraph (Milano)
Distribuzione: Sodip - Via Zuretti, 25 - 20125 Milano

Il Gruppo Editoriale Jackson è iscritto al Registro Nazionale della stampa al N. 117 Vol. 2 foglio 129 in data 17/8/1982.
Spedizione in abbonamento postale gruppo III/70
Aut. Trib. di Milano n. 102 del 22/2/1988

Il Gruppo Editoriale Jackson possiede per "Amiga Magazine" i diritti di traduzione di **Compute!'s Amiga Resource, Compute! Publications Inc. e Transactor for the Amiga, Transactor U.K.**
Amiga Magazine è una rivista indipendente non connessa alla Commodore Business Machine Inc., né con la Commodore Italiana S.p.a.- C64 e Amiga sono marchi registrati dalla Commodore Business Machine.

Associato al

Mensile associato all'USPI Unione Stampa Periodica Italiana

Consorzio Stampa Specializzata Tecnica

Testata aderente al C.S.S.T. non soggetta a certificazione obbligatoria per la presenza pubblicitaria inferiore al 10%

Il Gruppo Editoriale Jackson pubblica anche le seguenti riviste:

Amiga Magazine Games - Pc Games - Guida Videogiochi
Supercommodore 64 e 128 - Pc Software - Fare Elettronica
Bit - Computer Grafica & Desktop Publishing
Informatica Oggi - Informatica Oggi Settimanale
Pc Floppy - Pc Magazine - Automazione Oggi
Trasmissioni Dati e Telecomunicazioni - Elettronica Oggi
EO News settimanale - Strumenti Musicali - Watt
Meccanica Oggi - Strumentazione e Misure Oggi

Editoriale

Bentornati !!

Dopo aver passato il meraviglioso periodo di vacanze, eccoci tornati con un altro favoloso super-numero !

Lo speciale di questo mese è dedicato a Hollywood ! Vediamo come Amiga è diventata una star.

Nella parte hardware, questo mese, abbiamo fatto una prova pratica della Bridgeboard Commodore.

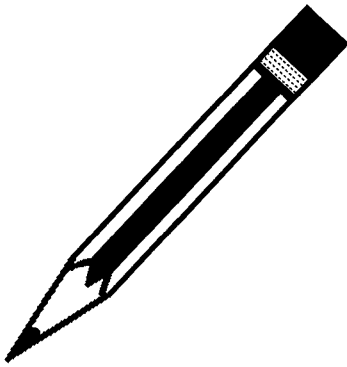
Nella parte software ci soffermiamo su: Superbase Professional, un sistema per gestire database, Virus Protection Toolbox, il pacchetto anti-virus più completo, M, un super-programma musicale e, infine, Pen Pal, un word processor con molte sorprese.

Per l'attualità, ovvero "Amiga Magazine in giro per il mondo", vi presentiamo le novità dell'AmiExpo di Washington (l'autore di questo articolo, è un nostro nuovo collaboratore e svolge, per Amiga Magazine, l'autorevole compito di corrispondente dagli U.S.A. !), Expoedit Copia, Amiga per il progetto Columbus e Basilea '90.

Inoltre, il nostro tecnico è andato alla scoperta delle 8520; continuiamo il "discorso" sul MIDI e, per i principanti ed ex possessori di C64, pubblichiamo un articolo dal nome molto esplicito: "Dal 6502 al 68000".

Per questo mese ci congediamo e... non dimenticatevi che c'è anche l'inserto di TRANSACTOR e ON DISK con il favoloso disco in cui troverete oltre 15 programmi !!

Arrivederci in edicola !!

La redazione

Compilatore

Spett.redazione,
scrivo questa lettera con la speranza di avere risposta al mio problema. Sono in possesso della versione 4.00 del compilatore C della Lattice. Quando mando in esecuzione il compilatore, da voi chiamato COMP nel libro "Linguaggio C" per Amiga 500, il file esegue il comando LC1, poi LC2 e quando arriva a BLINK si verifica un errore:
Error 426: Cannot find object
Blink failed returncode 140.
Questo errore sta tenendo a freno la mia voglia di imparare il C e per il momento sto apprendendo solo teoricamente.
Vi prego di rispondere al più presto al mio problema.
Distinti saluti

Baglione Massimo

Ti ringraziamo per i complimenti alla rivista e passiamo subito alla tua domanda.
L'errore che ti "assilla" è dovuto al fatto che il libro "Linguaggio C" è stato scritto quando sul mercato era presente la versione 3.40 del compilatore Lattice C.
Nella release 4.00 sono cambiate moltissime cose, tra cui anche il file lstartup.obj che è stato sostituito dal file c.o.
Di seguito trovi il file COMP aggiornato per la versione 4.00. Si assume che i file include si trovino in INCLUDE:, le librerie in LIB: e il compilatore e il linker in LC:, così come suggerito dal manuale Lattice.

```
; ***************************************
;
; File batch per la compilazione
;
; Versione per Lattice C  4.00 *
;
; ***************************************
;

.key prg
stack 20000
if not exists <prg>.c
   echo "File non disponibile !"
   Skip end
endif
echo "Inizio compilazione..."
LC:LC1 <prg>.c
if not exists <prg>.q
   echo "Errore di compilazione"
   quit 20
   endif
LC:LC2 <prg>.q
LC:Blink FROM LIB:c.o+<prg>.o
  LIBRARY LIB:
  lc.lib+LIB:lcm.lib+LIB:amiga.lib
  VERBOSE
delete <prg>.o
echo "Compilazione e linkaggio
terminati"
lab end
```

ATTENZIONE

Per motivi di spazio su disco non ci è stato possibile mettere il programma Post 1.1(Rubrica Public Domain). Ci scusiamo con i lettori e promettiamo che il suddetto programma sarà presente sicuramente nel numero 16 di Amiga Magazine

Note per l'uso dei programmi compattati

Il continuo sviluppo di programmi, sempre più di grande mole, ci vede costretti a dover compattare alcuni nostri programmi.
Ad esempio, nella directory Transactor e nella directory Utilities vi sono due programmi che sono stati compattati a causa della loro mole. I due programmi sono Scripit 1.20 e VLT e sono stati compattati con LHArc, un compattatore molto potente scritto da un nostro connazionale. Per poter decompattare, bisogna portarsi nella directory che contiene il file .LZH (questa è l'estensione che LHArc aggiunge ai suoi archivi) e digitare:

LHArc x nomefile.LZH <dest-path>

dove dest-path si indica un'altro drive, la RAM: o qualsiasi altro device che disponga di almeno mezzo Mega byte libero.
Ulteriori informazioni si ottengono digitando LHArc, che si trova nella directory C: del disco di Amiga Magazine 15.

International User Group

User Group del mondo unitevi !!

Questo è il motto che ha la nuova rubrica di Amiga Magazine. Se fate parte di un User Group e volete vedere pubblicato il vostro nome, inviateci i vostri dati: nome e il nome del vostro gruppo, indirizzo, telefono Bbs ecc.
Vengono pubblicati anche indirizzi di User Group stranieri in modo da aprire le frontiere fra utenti Amiga. La redazione non si assume nessuna responsabilità sui dati inviati e pubblicati.

Il nostro indirizzo è:
Gruppo Editoriale Jackson
Amiga Magazine - Area Consumer
Rubrica "International User Group"
Via Pola, 9
20124 Milano

U.S.A.

Birmingham Commodore Computer Club (BCCC), P.O. Box 59564, Birmingham, AL 35259-9564

Scottsboro Commodore Users Group, Rt. 5 Box 255, Scottsboro, AL 35768

Coosa Valley Commodore Club (C.V.C.C.), P.O. Box 1893, Gadsen, AL 35902 - 1893

Montgomery Area Commodore Komputer Society (MACKS), P.O. Box 210126, Montgomery, AL 36121 - 0126

Amiga/Commodore Club of Mobile, 3868 - H Rue Maison, Mobile, AL 36608

Commodore Mobile Users Group (CMUG) P.O. Box 9524, Mobile, AL 36691 - 0524

Anchorage Commodore Users Group (ACUG), Box 104615, Anchorage, AK 99510

Sitka Commodore User's Group, P.O. Box 2204, Sitka, AK 99835

La mostra di Washington (con qualche notizia da NY)

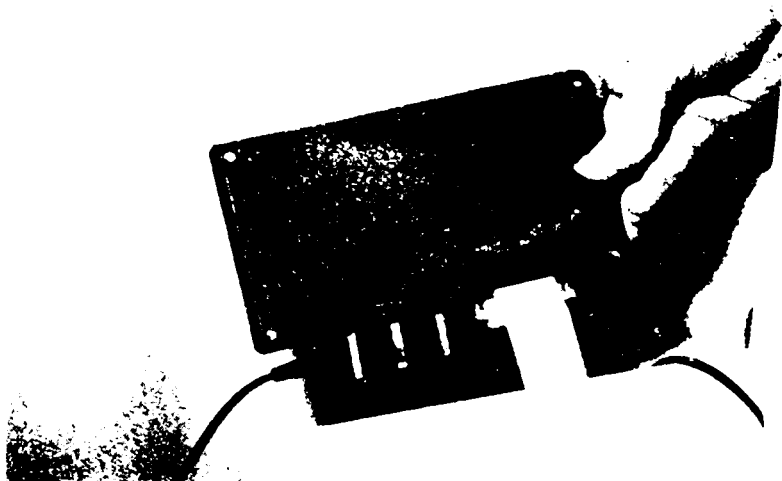
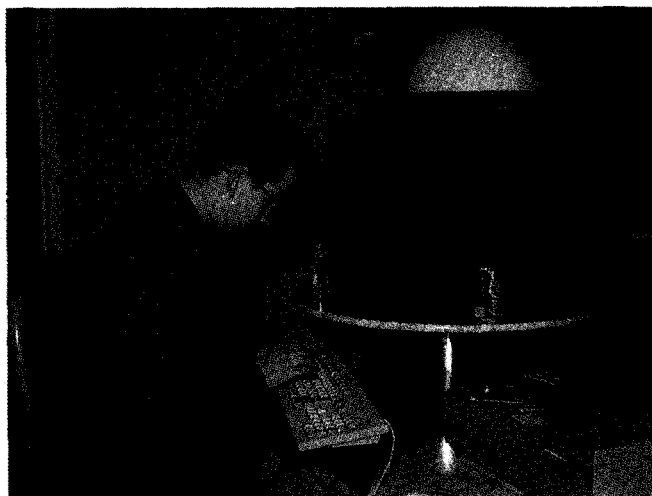
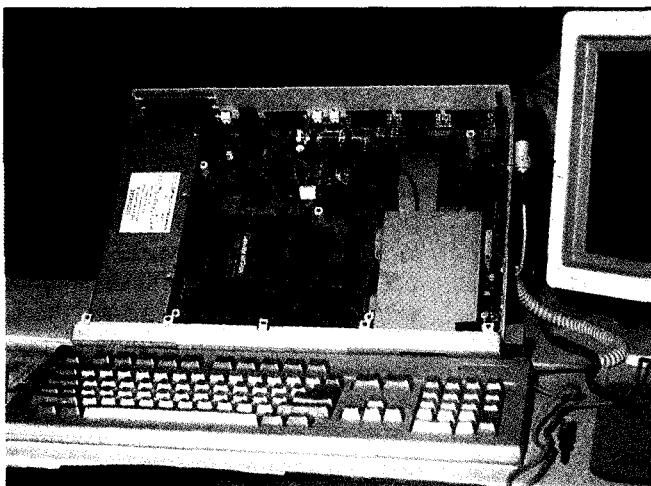
Il nostro corrispondente americano ha visitato, per tutti i lettori di Amiga Magazine, l'AmiExpo di Washington.

Marshal M. Rosenthal
Photo/Marshal M. Rosenthal NYC

Washington D.C. è la capitale degli Stati Uniti e possiede più viste meravigliose di quante si possano immaginare: monumenti, musei, parchi e, ovviamente, la sede del governo.
Ma noi non siamo venuti fin qui a vedere i ciliegi in fiore: la nostra visita all'hotel Sheraton, fuori dalla Connecticutt Avenue è troppo seria per indulgere in tali frivolezze (d'accordo non seria a tal punto, l'AmiExpo avviene qui e noi vogliamo vedere le ultime novità per i nostri computer). L'ingresso nella Showroom è, come sempre, una lotta.
Qui sfilano davanti ai nostri occhi valanghe di stand pieni di meraviglie. Dicono che i colori brillanti attirino l'attenzione, forse non è vero, dato che la prima cosa che sentiamo è la musica. Lo stand della Dr. T's si muove al suono del loro Phantom SMPTE Video, una combinazione di un sincronizzatore SMPTE e un MIDI Thru. Progettato per applicazioni professionali (SMPTE invia un segnale di sincronismo che permette alla musica di essere precisamente in sincronismo con una traccia audio o video), Phantom consiste sia di un hardware, che di un software. L'hardware può essere usato per il MIDI, e il prezzo contenuto di un'unità diviene persino più attraente quando consideriamo il costo per aggiungere un'unità MIDI ad Amiga. L'hardware sembra essere ovunque, e studiato per gli usi più inaspettati. Tra le solite versioni 500 e 2000, vediamo poche unità 1000. Questa vecchia macchina, ai sui tempi, era una bellezza e molti dei suoi possessori originari continuano a lavorarci attivamente.
Mentre la Commodore sta cercando in tutti i modi di eliminare la versione 1000 (offrendo vantaggiosamente la versione 2000 qui negli States), il nuovo hardware che sta uscendo, fa del 1000 una copia dei suoi più nuovi e più costosi fratelli. Il progettista Greg Tibbs ha creato il REJUVINATOR, una cartuccia che si inserisce nel 1000 e gli fa fare cose meravigliose. E' un Mega di memoria

Dr. T's SMPTE BOX

La Cart Rejuvinator

Show Maker Program

RAM on board che complementa un sistema che permette di usare i più nuovi sistemi ROM, l'Extended Chip Set (potete usare i 256K interni del WCS come memoria aggiuntiva avendo ora un Mega di RAM Chip e 512K Fast). E' incluso un orologio con batteria tampone e, come nell'A2000, una videoslot valida per usare il flicker fixer. In più potete scegliere tra il Kickstart interno e un disco esterno. Tutte connessioni sono fatte tramite cavi ad innesto, senza bisogno di alcuna saldatura (ma dovete fare molti disassemblaggi per inserire la cart).

Ci allontaniamo e torniamo nella confusione che ci porta all'AUTO SCRIPT, un tool professionale che vale la pena di esaminare.

Uno dei vantaggi delle immagini del PostScript è che vengono stampate in HI-RES e che non riportano nessuno dei soliti problemi grafici.

Script può importare file grafici del PostScript molto complessi e accurati, direttamente in programmi del tipo di Sculpt4D o Turbo Silver, permettendo l'uso di pacchetti di alta qualità come Aldus Freehand, LetraStudio, più Gold Disk's Amiga-Based Professional Draw. Il tutto in un modo semplice e intuitivo, può lavorare con file MS-DOS, ma non significa che debba avere una Command Line! Più cose aggiungi ad un sistema, più hai la possibilità che qualcosa vada storto. Meglio prevenire che curare, così sarebbe il massimo portare il computer dal dottore, questo è il DOCTOR AMI.

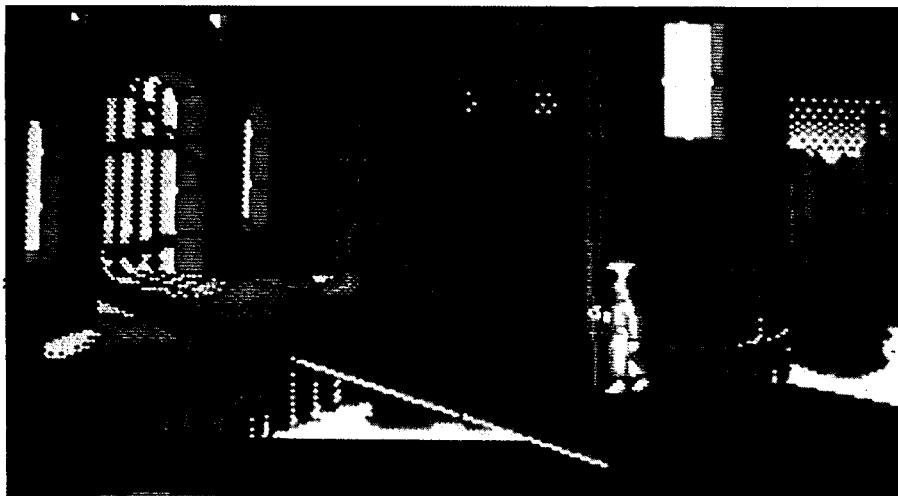
Si tratta di due programmi: Memory Doctor fa un controllo di tutto il sistema e delle espansioni di memoria per localizzare e identificare difetti e chip difettosi. Inoltre, mappa le locazioni di memoria guaste.

La seconda parte, Drive Doctor, controlla tutti i settori del disco per cercare, identificare e mappare eventuali errori hardware e software. Ciò che suscita molta attenzione è il nuovo Video Toaster della New Tek. Tenete presente che il Toaster è stato annunciato dalla New Tek da oltre 4 anni, con promessa della sua uscita con regolarità ad ogni fiera.

Ora l'unità è uscita ed è molto interessante per ciò che può fare assieme alle potenzialità del multitasking di Amiga. Comandato da software, il Toaster consiste di una cartuccia che viene inserita nelle versioni 2000-2500-3000 (sono consigliati 3 Mega di RAM). Contiene un proprio set di chip custom che rendono possibile una risoluzione di qualità broadcasting (standard U.S.), con tutti i 16,8 milioni di colori necessari. Tutto viene controllato da un'interfaccia "punta e clicca", l'input/output è collegato direttamente alla cartuccia che ha quattro in perput video esterni, due canali digitali e un generatore di disegni. Due buffer rendono le immagini in 3D a 24 bit e i disegni possono essere fatti con il software opzionale che uscirà presto. Il caricamento delle immagini è veloce, sia dall'hard disk che dalla memoria RAM. Un banco di commutazione permette la dissolvenza, gli effkey e color, mentre un preview output vi permette di controllare ciò che ciascuno sta per vedere. Gli effetti digitali video avvengono in tempo reale, con immagini che girano, vengono schiacciate, zoomate eccetera. C'è anche il processo per creare l'effetto solarizzazione e quello dei filtri fotografici; un generatore di caratteri con scroll incorporato e una moltitudine di font e opzioni, oltre ad un "cattura" di immagini per salvarle (frame grabber). Forse l'effetto più incredibile è il prezzo, circa 1600 dollari USA, davvero pochissimo per avere tutte queste prestazioni in una sola scatoletta !

Una volta passata questa meraviglia, ci troviamo davanti a file di ditte che ci aspettavamo e che mostrano i loro prodotti. Inaspettata, invece, l'Electronic Arts; è la sua prima presenza ad una mostra di Amiga.

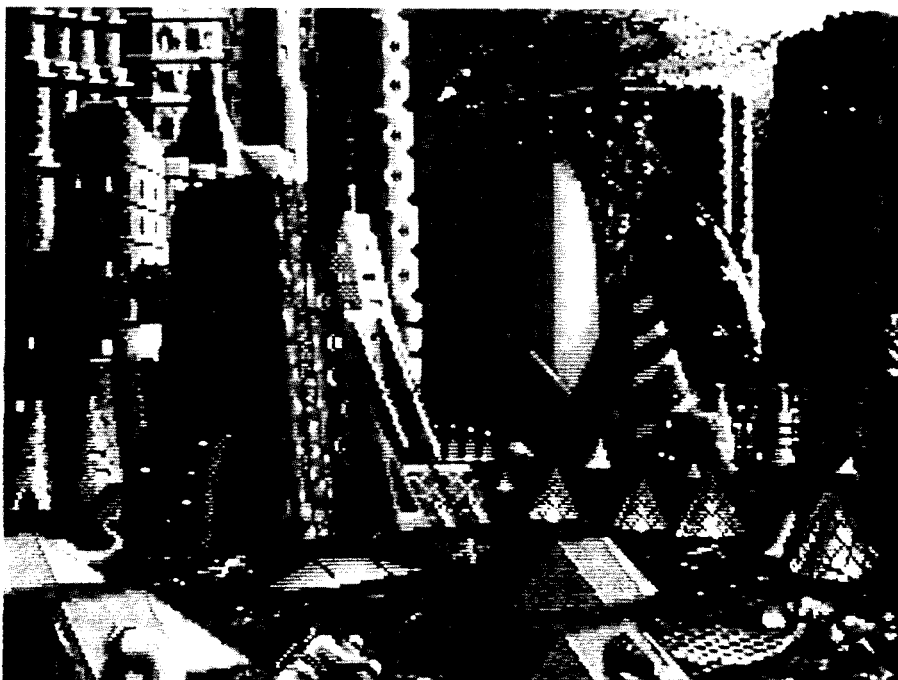
Oltre a mostrare il Deluxe Video III, presenta altri tre marchi affiliati.

La Lucasfilm presenta LOOM, il loro

nuovo gioco grafico/avventuroso, pieno di fantasia e musica. Offrono anche Battle of Britain, con effetti grafici che sono possibili grazie alla qualità grafica superiore di Amiga, con cui attaccate il nemico dalla cabina del vostro bombardiere. Alcuni programmi particolari fanno parte di questi eventi, e la Gold Disk mostra uno spettacolo multimedia attraverso la presentazione del loro ShowMaker, di prossima uscita. Piuttosto che seguire la folla, noi ci infiliamo nella vicina hospitality suite per poterlo vedere meglio, grazie alla cortesia del direttore della Gold Disk, David Jones. Egli comincia: "MULTIMEDIA è l'espressione degli anni ottanta, e sarà disponibile nei novanta". Lo ShowMaker rende il desktop video multimedia una realtà. Permette l'integrazione di animazione, della grafica, del MIDI, del suono del computer, del disco laser, Genlock controllati dal software e altre fonti di immagini video. Studiato per applicazioni professionali, ma con un mezzo popolare, ShowMaker ha una funzione di auto-load che funziona durante gli eventi, così il programma guarda sempre avanti (ciò richiede come minimo 1 Mega di memoria RAM e un hard disk per essere effettivo). E' compresa la titolazione, come una quantità di possibilità video, quali wipe e il fade.
"Credo che l'aspetto più importante del programma sia la possibilità di sincronizzare automaticamente la musica e il video" fa notare Jones "questo elimina oggi un importante problema nel desktop video, e vi permette di definire un'animazione in termini di frame per battuta musicale, invece di frame per secondo". Ce lo dimostra assemblando un breve pezzo, usando la grafica statica IFF, un segmento di animazione, più la musica da una tastiera MIDI. Un'interfaccia molto semplice e intuitiva mette ogni cosa insieme sullo schermo di Amiga in un modo logico e semplice. Il programma mostrerà e stamperà degli storyboard e persino bozze del lavoro in corso. Altre caratteristiche includono un clock esterno per l'input MIDI, e l'attesa per il comando di interazione tra le varie periferiche e Amiga. Le mostre di computer sono come i pasti, alcuni pesanti, alcuni leggeri. Garantito che i giochi si vendono sempre bene, è un segno positivo vedere ditte che investono in periferiche hardware. Significa che vogliono fare un investimento extra nel futuro di Amiga e aspettarsi un ritorno finanziario. Tutto ciò può essere tradotto in una lunga e sana vita per il nostro computer preferito.

Dati sull'autore:

Marshal M. Rosenthal ha passato una vita nell'industria dell'intrattenimento elettronico.
Giocare con i videogame e sviluppare l'Hi-tech può essere faticoso, ma egli è perfettamente convinto di tirare avanti così (heh-heh). ▲

Loom

Battle of Britain

Articolo tradotto dal n.3 Fall Edition 1989 di Compute!'s Amiga Resource.

Un ponte sull'abisso

Una prova pratica della Bridgeboard della Commodore

Neil Randall

Amiga 1000 fece il suo debutto nel 1985, quando il PC IBM era il solo computer professionale preso sul serio. Il Macintosh, con un solo anno di vita, era ancora considerato come un giocattolo dalla folla dei professionisti, e l'Apple II e il Commodore 64 non erano stati fatti per il lavoro. Non può meravigliare, allora, che uno dei maggiori obiettivi della Commodore, nei confronti di Amiga, fosse la compatibilità con l'IBM. E dal momento che la famiglia degli IBM e la sua crescente legione di cloni, girava sotto l'MS-DOS della Microsoft, allora anche Amiga doveva diventare un clone MS-DOS.

Dapprima la Commodore pensò di risolvere il problema via software. Nel 1986 la compagnia rilasciò Transformer, che, assieme a un disk drive da 5.25", il Commodore 1020, portava di fatto le capacità dell'MS-DOS su Amiga. Ma emersero ben presto tre problemi. Primo, il Transformer permetteva di far girare i pacchetti mostro dell'MS-DOS, come WordPerfect, WordStar, dBase II e Lotus 1-2-3, ma aveva problemi con molti altri pacchetti, specialmente quelli orientati, in qualche modo, alla grafica (una grande ironia, considerando le capacità grafiche enormemente superiori di Amiga). Secondo, Transformer era lento, girando ben al di sotto dei 4.77 MHz raggiunti persino dai PC più disadorni. Infine, gli utenti MS-DOS si erano rapidamente abituati agli hard disk e il mercato Amiga non era ancora abbastanza esteso da sostenerli.

Il software Transformer è ancora disponibile, e anche programmi di public domain che lo velocizzano. Ma la vera compatibilità con l'MS-DOS è realizzata dall'hardware, non dal software, perché gli Amiga e i PC non usano solo differenti microprocessori, ma anche microprocessori prodotti da compagnie del tutto diverse. Gli Amiga (come il Macintosh e l'Atari ST) usano il chip 68000 della Motorola, mentre i PC usano l'Intel 8086 o 8088. Farli sposare non è un compito facile.

Infine, la Commodore sviluppò il Sidecar, un modulo hardware aggiuntivo per Amiga 1000, che era, in effetti, un PC XT, IBM compatibile, in un piccolo contenitore separato. Quando la compagnia decise di fermare la produzione del 1000 per passare al 500 e al 2000, abbandonò anche il Sidecar. Al suo posto la Commodore introdusse la Bridgeboard A2088, una scheda interna per Amiga 2000.

Come funziona?

La Bridgeboard A2088 è una scheda hardware che si colloca in uno dei due slot speciali che risiedono nel 2000. Siccome si colloca fisicamente a cavallo degli slot PC e Amiga, funziona precisamente come il ponte (bridge) che il suo nome suggerisce. Con un'impressionante impresa di ingegneria hardware, la Commodore è riuscita a mettere in comunicazione due mondi informatici completamente diversi.

Ciò che la Bridgeboard deve fare, è permettere a una macchina MS-DOS di funzionare contemporaneamente ad Amiga, e come sua parte. A tal fine la Bridgeboard contiene sia un computer separato, sia circuiti che permettono a due computer di

condividere un sistema. Ciò significa, dal punto di vista funzionale, che il processore Intel 8088 della Bridgeboard, il cuore dei cloni PC XT IBM, deve lavorare congiuntamente con un Motorola 68000, la CPU di Amiga.
Far lavorare l'8088 non costituisce, di per sé, un problema. La Commodore aveva da due anni una linea di computer PC compatibili, così la tecnologia era già pronta. Il PC della Bridgeboard funziona come un puro XT, con un 8088 a 4.77 MHz, uno zoccolo per il coprocessore matematico 8087, 512K di RAM, un BIOS da 16K progettato dalla Phoenix Tecnologies e un disk drive da 5.25" (con controller su scheda) destinato a inserirsi nell'apertura inferiore del 2000. Per la Bridgeboard la tastiera Amiga emula una tastiera XT standard, mentre la porta parallela (per la stampante) può essere assegnata al lato PC. Mentre tutto questo è in funzione, il monitor Amiga emula o il modo colore o quello monocromatico dell'IBM.
Il resto dell'emulazione PC è assicurato da slot e porte. La stessa Bridgeboard contiene un connettore parallelo Centronics, nel quale si può inserire un disk drive esterno. Possono esservi collegati sia il Commodore 1020, un drive Amiga da 5.25", sia lo standard Amiga 1010, un drive esterno da 3.5", capace di formattare un dischetto a doppia faccia con la capacità di 720K. Non appena inserite la Bridgeboard nello slot più a destra del 2000, avrete a disposizione tre slot PC per periferiche, come adattatori video, schede per hard disk, schede musicali, schede acceleratrici e persino schede di emulazione AT.
Pur essendo separati, il lato PC e quello Amiga condividono 128K di RAM molto importante, chiamata "dual-port" (a doppio accesso). Attraverso la RAM dual-port, che è suddivisa in tre sezioni, Amiga può interagire e controllare il PC XT che risiede sulla Bridgeboard. Assieme a due grandi chip custom della Bridgeboard, che contengono la configurazione di controllo di Amiga e l'interfaccia per il software Janus, la RAM dual-port permette ad Amiga di visualizzare e controllare il PC XT, come se fosse semplicemente un'altra finestra Amiga.

Mettiamola al lavoro

1. Il montaggio

Imparare a usare la Bridgeboard non è immediato. Si può prevedere con sicurezza che chiunque compri un Amiga 2000, senza averne mai usato uno, e vi aggiunga una Bridgeboard senza avere mai usato l'MS-DOS, di fatto non sarà in grado di iniziare. Quelli che conoscono un sistema, ma non l'altro, avranno una vita in qualche modo più facile. L'ideale, tuttavia, è conoscere sia l'AmigaDOS che l'MS-DOS o almeno trovare qualcuno che prepari il sistema per voi.
Per caricare l'MS-DOS su Amiga, dovete dapprima montare la Bridgeboard. Il caso migliore, in questo caso, è che il rivenditore la installi per voi, soprattutto per il fatto che l'apertura del contenitore del 2000 può annullare la vostra garanzia. Installare la Bridgeboard, tuttavia, è piuttosto facile: semplicemente si apra il 2000 e si seguano le istruzioni del manuale Bridgeboard. In tutto, il lavoro richiede circa mezz'ora.

2. Monocromatico o a colori, e chi gestisce la stampante?

Quando Amiga è stata rimontata, inserite il disco di sistema dell'MS-DOS (incluso nella Bridgeboard) nel disk drive da 5.25" (che avete montato con la Bridgeboard) e accendete Amiga. Quando Amiga richiederà il disco Workbench, inserite il disco PC-Workbench, incluso anch'esso nel pacchetto della 2088. Questo disco contiene una Startup-Sequence modificata che lancia l'MS-DOS dopo avere caricato il Workbench. Infine, dovete solo aprire una finestra MS-DOS.
Per aprire una finestra MS-DOS, selezionate due volte il drawer PC che si trova sul disco Workbench. Apparirà una finestra con i programmi "PC Mono", "PC Color", "PCDisk", "LPT1" e "PCPrefs". "PC Mono" apre uno schermo monocromatico MS-DOS, mentre "PC Color" fornisce uno schermo a colori. Selezionando "LPT1" si assegna la porta per la stampante di Amiga al lato PC. "PCDisk" lancia il software necessario per trasferire file tra l'MS-DOS e l'AmigaDOS e per usare i disk drive di Amiga come dischi virtuali (si veda oltre). "PCPrefs" vi dà la possibilità di disabilitare lo schermo Amiga (utile se comprate un monitor separato per il lato PC) e vi permette di stabilire la quantità di memoria che il vostro sistema userà per l'interfaccia Amiga/Bridgeboard.
Un esempio della difficoltà nell'uso del software della Bridgeboard è lo sforzo necessario per aprire uno schermo PC a colori. Al momento del boot lo schermo di default per la Bridgeboard è monocromatico; pertanto uno schermo a colori deve essere richiesto in maniera specifica. Per fare questo, selezionate due volte l'icona "PC Mono"; poi digitate "MODE CO80" sullo schermo MS-DOS e premete RETURN. Poi dovete chiudere la finestra "PC Mono" (selezionando il gadget di chiusura) e selezionare due volte l'icona "PC Color" nella finestra "PC-Workbench". Ora potete modificare lo schermo di default, rendendolo a colori, ma il punto è che, per gli utenti iniziali, tutto ciò è insieme sconcertante e non necessario. Senza dubbio, sarebbe bastato un requester che chiedesse se si desidera uno schermo monocromatico o a colori al momento del boot.

3. Guardando attraverso la finestra PC

Se si tiene presente che la finestra PC è proprio come qualsiasi altra finestra Amiga, non è sorprendente scoprire che essa contiene i menu pull-down, tipici di Amiga. In primo luogo, questi menu servono per configurare la finestra PC come si desidera.
Potete scegliere "Hide Borders" per eliminare i bordi della finestra (in modo che assomigli di più allo schermo di un PC), oppure scegliere "Small-Size Window" per restringere la finestra e porla in un angolo (al fine di accedere al Workbench). "Window Freeze" vi permette di mantenere il contenuto di una finestra PC così com'è (di solito per lavorare contemporaneamente su un'altra finestra PC), mentre "Set Cursor Blink Rate" vi permette di modificare l'intervallo con cui lampeggia il cursore. Inoltre, potete aprire una seconda e una terza finestra PC (in multitasking), o stabilire il numero di colori del testo (massimo 4 con schermo monocromatico, 16 con quello a colori). A differenza di un PC reale, il PC di Amiga può scegliere i suoi colori fra i 4096 disponibili sulla macchina ospite.
E' anche inclusa una rudimentale funzione di taglia-e-incolla. Questa vi permette di trasferire testi tra le finestre PC e altre applicazioni di Amiga. Sfortunatamente, andrete incontro a dei problemi se la vostra applicazione per Amiga non usa la clipboard standard (WordPerfect è il primo a non usarla).
Tuttavia, questo non è un difetto della Bridgeboard; questi programmi, che trasgrediscono le regole, non possono fare operazioni di "taglio", "copia" o "incolla" a nient'altro che a se stessi.

4. Virtuale o reale? Solo Janus lo sa con certezza.

La Bridgeboard si presenta con un disk drive da 5.25", che va installato dentro Amiga 2000. Esso formatta i dischi a doppia faccia con una capacità di 360K. Il problema è che molti programmi PC richiedono due drive e che un numero sempre crescente è inutilizzabile senza un hard disk. Nel mondo PC, si sa, gli hard disk sono a buon mercato.
Ci sono cinque modi per aggiungere un drive in più alla Bridgeboard:
* Comprare un drive da 5.25" aggiuntivo e inserirlo nella porta per drive fornita dalla Bridgeboard. Il Commodore 1020 è perfetto per questo scopo, ma molti altri drive sono disponibili sul mercato.
* Comprare un drive esterno da 3.5" e inserirlo nella porta della Bridgeboard. Se già lo possedete, andrà bene il Commodore 1010 (il normale drive di espansione di Amiga). Di fatto, potete staccarlo dal connettore della Bridgeboard e collegarlo a quello di Amiga secondo le necessità.
Si noti, tuttavia, che la Bridgeboard formatterà il disco in questo drive con solo 360K di capacità, a meno che voi non modifichiate il file CONFIG.SYS sul vostro disco di sistema MS-DOS. Per far ciò, caricate CONFIG.SYS in un buon word processor (non usate l'inguaribilmente ostico programma EDLIN, fornito con l'MS-DOS); poi aggiungete al file la linea:
DRIVPARM=/D:01 /F:2 /H:2 /S:9 / T:80
assicurandovi di inserire tutti gli spazi, e salvatelo in formato ASCII (testo). A questo punto effettuate un reboot dell'MS-DOS, premendo i tasti CTRL, ALT e DEL contemporaneamente. Potete ora formattare i dischi da 3.5" con una capacità di 720K.
* Comprare un hard disk con controller per Amiga. Il controller Commodore 2090A costituisce lo standard (non comprate il 2090, perché non permette di effettuare l'autoboot), inoltre esistono altri hard disk sul mercato. Il lato MS-DOS potrà condividere questo drive, usandolo come drive virtuale (si veda l'opzione finale in questa lista).
* Comprare un hard disk con controller per MS-DOS. Il lato Amiga potrà condividere questo drive usando il software Janus fornito con la Bridgeboard. Questi drive si chiamano drive Janus.
Si noti che la scelta tra la terza e la quarta opzione dipende da quale sistema usate più frequentemente. Un hard disk Amiga vi permetterà di effettuare l'autoboot al momento dell'accensione (non avrete bisogno del disco Workbench) e vi permetterà un accesso più veloce al disco dal lato Amiga, piuttosto che da quello MS-DOS.
Modificando la Startup-Sequence di Amiga, potete anche effettuare l'autoboot dell'MS-DOS, usando il drive. I soli benefici che provengono dall'acquisto di un hard disk MS-DOS sono l'aumento di velocità dal lato MS-DOS e il prezzo molto ridotto. Si calcolino 800.000 lire per un drive Amiga da 40 Megabyte e 400.000 lire per un drive MS-DOS con la stessa capacità.
* Usare i drive Amiga come drive virtuali MS-DOS. La combinazione dell'icona "PCDisk" sul disco PC-Workbench e del programma "JDISK" sul disco MS-DOS vi consente di usare i drive Amiga come se fossero drive MS-DOS. Ciò significa che potete usare i drive interni del 2000 (df0: o df1:), il drive esterno di Amiga (df2:) o l'hard disk di Amiga per il lato MS-DOS del sistema. Creare dei drive virtuali non è particolarmente difficile, ma il manuale della Bridgeboard, che per altri versi appare molto ben fatto, si dilunga fino a generare molta confusione sul procedimento da usare.
Quando create un drive virtuale, non state facendo altro che dire al drive di Amiga di riservare dello spazio su disco per l'MS-DOS.
Tecnicamente, questo spazio è semplicemente un file AmigaDOS che Amiga può leggere, ma che, dal punto di vista dell'utente, appare a tutti gli effetti un disco MS-DOS. Il problema è che il software JLINK non è affatto facile da implementare.

Procedere con cautela

La procedura che segue vi dice esattamente come fare per creare un drive virtuale MS-DOS sul drive interno del 2000 (df1:). Sono disponibili molte opzioni, ma dovreste esplorarle solo dopo aver messo a punto la procedura che segue.

* Il file CONFIG.SYS deve contenere la linea:

DEVICE=JDISK.SYS

Se così non fosse, usate il vostro word processor preferito per aggiungere questa linea ed effettuate un reboot dell'MS-DOS premendo i tasti CTRL-ALT-DEL.
* Aprite il PC-Workbench e selezionate due volte "PCDisk" (si noti che dovete farlo o non potrete usare il drive virtuale).
* Aprite una finestra MS-DOS selezionando due volte l'icona "PC Mono".
* Rispondete alle richieste di ora e data e giungete al prompt A>.
* Ponete un disco formattato per Amiga (preferibilmente vuoto) in df2:.
* Cambiate con il drive A: e digitate:

JLINK D: DF2:VIRTUAL /C:160

* Ora usate il vostro software MS-DOS. Quando desiderate usare il drive virtuale per caricare, salvare, copiare e così via, dite al programma di usare il drive D:.
* Quando avete finito di usare l'MS-DOS, non chiudete la finestra e non spegnete la macchina. Invece, scollegate il drive virtuale digitando:

JLINK D: /U

(se non fate così, potreste danneggiare il file da voi creato).
* Quando volete usare di nuovo il drive virtuale, fate il boot dell'MS-DOS, ponete lo stesso disco in df2: e digitate:

JLINK D: DF2:VIRTUAL

* Ancora una volta potete caricare e salvare file in df2: attraverso D:.
Qualche spiegazione. Siete obbligati a chiamare il vostro drive virtuale D:. JLINK permette C:, D:, E: o F:. Infatti, potete avere quattro drive virtuali che funzionano contemporaneamente. Inoltre, potete specificare qualsiasi drive Amiga, come destinazione del vostro drive virtuale; assicuratevi solamente di avere in esso un disco formattato. Per esempio, JLINK C: DH0:VIRTUAL porrà un drive virtuale chiamato C: sull'hard disk di Amiga, emulando con ciò il normale sistema hard disk dell'MS-DOS. Potete anche creare drive virtuali nel RAM disk di Amiga (JLINK E: RAM:VIRTUAL), nel nuovo disco RAD, in df0: o in df1:.
Infine, l'ampiezza che attribuite al drive virtuale (/C:160) non è molto importante, come avviene per il RAM disk di Amiga, il drive virtuale crescerà man mano che i dati vi vengono immagazzinati. E' per questo che ho raccomandato di usare un disco vuoto come disco virtuale: non vorrete certo che il vostro drive virtuale debba combattere con i file di Amiga per lo spazio su disco.

Funziona veramente?

Si funziona veramente. Installata e configurata correttamente, la Bridgeboard è un clone dell'IBM PC XT. Ho lanciato molti programmi MS-DOS sulla Bridgeboard e hanno lavorato tutti senza problemi.
Ho sentito dire che alcuni giochi, che fanno intenso uso della grafica, non funzionano correttamente, ma tutti quelli che ho provato: Starflight, Star Commander, Flight Simulator II, Nobunagàs Ambition, Civil War, Pete Rose Baseball, Heroes of the Lance, non hanno avuto alcun problema. Analogamente, ho lavorato con successo con Word Perfect, Microsoft Word, Microsoft Works, Lotus 1-2-3 e altri pacchetti applicativi. La scheda AdLib's Music Syntetizer ha funzionato perfettamente in uno slot PC e ho parlato con persone che hanno usato senza problemi schede VGA e schede per hard disk. Il manuale della Bridgeboard promette totale compatibilità con schede quali la AST Six Pak Plus, l'ATI EGA Wonder, la scheda d'espansione Intel Above Board 2MB, il modem interno Hayes Smartmodem 1200, la Plus Hardcard e la scheda Taurus LAN.
A 4.77 MHz, tuttavia, la Bridgeboard è lenta. La maggior parte dei cloni PC XT ora monta processori turbo o a doppia velocità e i loro utenti noteranno la differenza. Un tecnico mi ha suggerito, ufficiosamente, di sostituire il processore 8088 della Bridgeboard con il più veloce 8088-2, ma non ho ancora avuto il coraggio di provare.
Se optate per la Bridgeboard, comprate anche un hard disk. Se il vostro budget è un po' limitato, comprate un hard disk MS-DOS e preparate delle partizioni per usarle dal lato Amiga (il manuale spiega come). Se prevedete di usare software MS-DOS in maniera intensiva, comprate un drive aggiuntivo MS-DOS (i drive virtuali sono adatti solo per usi saltuari). Se non potete assolutamente permettervi un hard disk, comprate un drive esterno e inseritelo nello slot della Bridgeboard. Non vi aspettate risultati soddisfacenti da un sistema Bridgeboard con un solo drive.

Il presente e il futuro

Nel corso del 1989, la Commodore ha posto in vendita alla Bridgeboard A2286. Questa scheda si accosta la 2088 e dà ad Amiga piena compatibilità con gli AT IBM. La A2286 contiene un processore Intel 80286 (su cui sono basati gli AT) e un disk drive da 5.25" da 1.2MB. Questa versione della Bridgeboard gira a 8 MHz, molto più velocemente della 2088, ma ancora ben al di sotto dei 10-12 MHz tipici dei cloni AT.
La Commodore stima che un acquirente su due di Amiga 2000, compri una Bridgeboard.
E' impossibile prevedere se questo trend si manterrà anche con la più costosa A2286. Si può prevedere, comunque, un ulteriore aggiornamento verso l'80386 e oltre, perché la combinazione Bridgeboard/Amiga è molto potente.
L'interfaccia software deve essere migliorata drasticamente per portare il sistema ad esprimere tutte le sue potenzialità, ma avere due computer in un solo contenitore è un'idea molto buona.
Per coloro che vogliono Amiga, ma hanno anche bisogno di accedere all'MS-DOS, la Bridgeboard è una soluzione quasi ideale. ▲

A Pesca di suoni con Perfect Sound

Aldo & Andrea Laus

Questo mese affrontiamo un argomento di sicuro interesse per chi intende fare musica con Amiga, utilizzando i quattro generatori di suono di cui il nostro computer è dotato.
E' noto, infatti, che grazie ai suoi quattro circuiti digitali per riprodurre suoni campionati, contenuti nel chip PAULA, Amiga, tramite opportuni programmi, può svolgere le funzioni di quelle macchine musicali note con il nome di lettori di campioni e che normalmente costano come il computer stesso.
Questo in fase esecutiva, ma, la nostra macchina, con l'aggiunta di un piccolo dispositivo hardware e un altro tipo di programma, è anche in grado di effettuare registrazioni di suoni, voci singole o parecchi secondi di brani musicali, sotto forma di informazioni digitali.
Queste possono essere manipolate con l'aiuto di una rappresentazione grafica e poi archiviate sotto forma di file IFF per un successivo utilizzo con i programmi di esecuzione musicale.
E qui viene il bello, perché, a parte qualche eccezione che vedremo in seguito, Amiga ha il vantaggio di disporre di un sistema standard di file che li rende intercambiabili fra diversi programmi.
Quando Amiga era in fase di sviluppo, un gruppo di progettisti della Commodore e della Electronic Arts (nota software house che ha da sempre supportato Amiga con pacchetti di qualità quali Deluxe Music Construction Set, che tutti conosceranno e che rappresenta un buon modo per iniziarsi alla musica gestita dal computer) hanno introdotto uno standard per definire il formato dei file in modo da poter facilmente scambiare i dati fra diversi programmi.
Tale standard è stato appunto battezzato IFF (Interchange File Format) e definisce delle strutture di dati per gestire, ad esempio, immagini (file ILBM), spartiti musicali (file SMUS), testi (file FTXT) e suoni campionati a 8 bit (8SVX).
Il tipo specifico di file viene riconosciuto dai programmi grazie a degli identificatori posti all'inizio che ne indicano i parametri essenziali per la loro gestione (tipo, lunghezza della sezione dati, fine della sezione dati, nome ecc..).
Non vogliamo qui entrare in dettagli tecnici sull'architettura dei file IFF, per i quali rimandiamo all'ampia letteratura esistente (ad esempio, l'articolo "Lo standard IFF" di Leonardo Fei sul numero 6/89 di Transactor per Amiga).
Il nostro obiettivo è invece di orientare gli appassionati di computer music alle applicazioni che i file IFF rendono possibili.
Come visto sopra, i file per eccellenza che interessano i musicisti sono di due tipi:

- l'8SVX che riguarda le voci audio da utilizzare con i programmi musicali o più semplicemente i suoni che vengono usati nei giochi.

Talvolta troviamo anche interi brani per la presentazioni di giochi. In

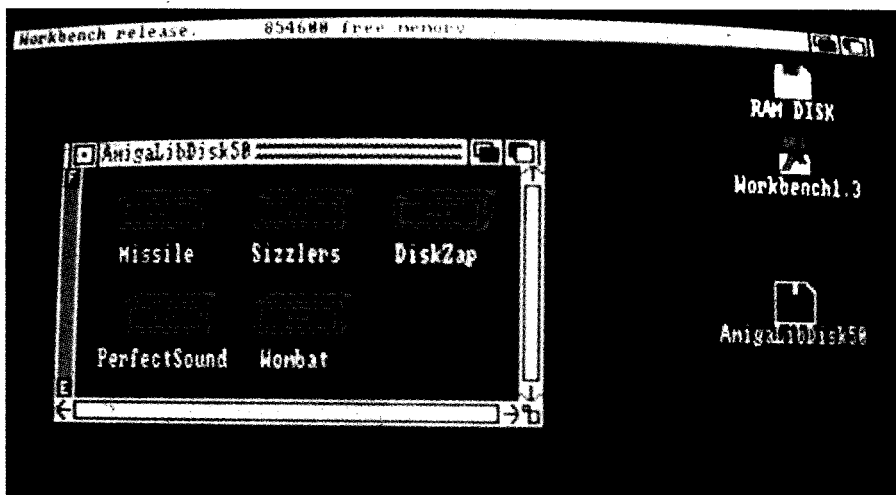
Schermata di presentazione del programma Perfect Sound

Il menu del Fish Disk 50

questi file sono codificati forme d'onda, inviluppo, volume ecc..., relativi a ciascuna voce per una sola nota.
Ci penserà poi il programma musicale a generare tutte le note necessarie al brano da eseguire.

• lo SMUS, invece, contiene le informazioni di brani musicali con le quali è possibile sia generare la notazione musicale sotto forma di righi di spartito sullo schermo, che di eseguire il brano indirizzando le note ai suoni disponibili.
Ovviamente in questi file sono codificate tutte le informazioni necessarie (tempo, numero di traccia, volume ecc..).

La generazione dei suoni di Amiga

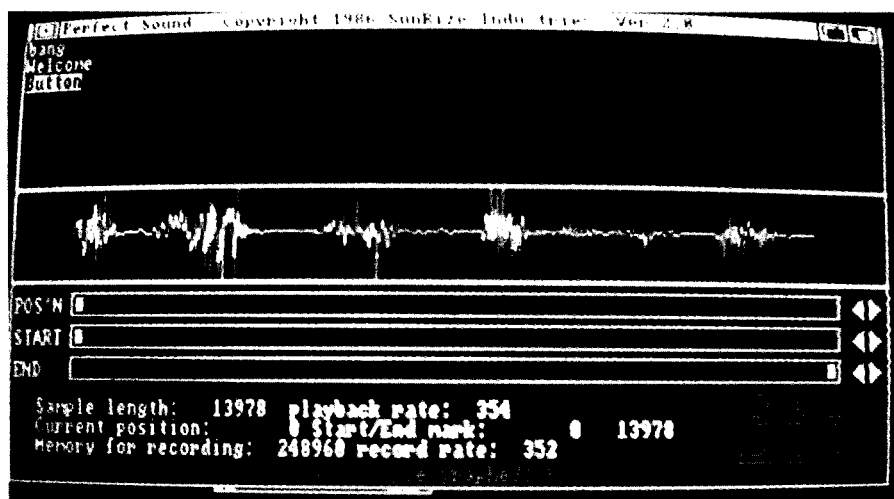
Amiga dispone di un hardware dedicato per produrre il suono.
Esso consente di generare contemporaneamente fino a quattro canali di suono, i quali vengono assegnati alle due uscite audio del computer.
La generazione dei suoni avviene tramite un processo digitale chiamato campionamento, analogo a quanto avviene per i compact disc, il DAT e i sintetizzatori musicali digitali.
Nel processo di campionamento, un'onda sonora analogica (la voce, una nota musicale o un brano) viene trasformata in una serie di dati digitali attraverso un dispositivo hardware chiamato campionatore (o digitalizzatore audio).
Questi dati sono convertiti, nel caso di Amiga, in un formato a 8 bit e memorizzati in file IFF del tipo 8SVX.
Quando un set di questi dati viene inviato ad un generatore di suono di Amiga, questi riproduce fedelmente il suono analogico originale.
I suoni possono essere modificati a piacere tramite opportuni programmi, intervenendo sulla loro frequenza, inviluppo, volume ecc., dando così la possibilità di realizzare archivi personalizzati.

Come procurarsi i dati dei suoni

Ci sono due metodi, fondamentalmente, per collezionare le voci con le quali poi eseguire i vostri brani musicali attraverso altri programmi:

1- Attrezzarvi con un digitalizzatore audio e uno o più programmi per il campionamento; con questa attrezzatura sarete in grado di catturare suoni dal vivo, da un registratore, dalla radio e da ogni altra sorgente sonora.

2- Procurarvi uno o più programmi campionatori e "andare a caccia di suoni" sui dischetti di programmi che già possedete.

Oggi esistono diverse proposte sul mercato sia per i digitalizzatori hardware, che per i relativi programmi.
Per chi si trova a cominciare e vuole farlo in forma economica, proponiamo di ricorrere ad un classico del campionamento, disponibile sul disco 50 della raccolta di Fred Fish: PERFECT SOUND della Sunrize Industries, nota azienda USA del settore.
Il disco suddetto può essere ordinato alla redazione Jackson che ve lo invierà secondo le modalità indicate nell'apposita pagina di questa stessa rivista.
Se vi interessa un digitalizzatore, noi abbiamo sperimentato quello offerto in kit da Fare Elettronica N.44 (e che riproporremo su uno dei prossimi numeri di Amiga Magazine) che funziona perfettamente assieme al programma PERFECT SOUND.
L'unico accorgimento suggerito, è di registrare i suoni in uscita da un registratore per avere un livello adeguato per pilotare il digitalizzatore.
Con questi semplici tool potete crearvi dei dischetti di voci IFF da impiegare poi con i programmi musicali.
Il procedimento è semplice ed è descritto di seguito.
Anche senza disporre del digitalizzatore, potete crearvi una raccolta di suoni con il Perfect Sound, utilizzando la funzione LOAD, "pescando", dai dischetti di giochi o di altri programmi musicali eventuali suoni presenti sotto forma di file IFF.
Una volta catturato un suono, questo può essere salvato subito sul vostro dischetto, oppure modificato a vostro piacere prima di essere salvato.
Mentre andate alla ricerca dei suoni, dovete fare attenzione, infatti non tutti i programmi adottano i file IFF standard.
Ad esempio, il noto programma musicale SONIX ha un proprio formato.
E' anche vero, però, che il famoso programma MUSIC-X contiene una utility che converte il formato SONIX in IFF (ma non viceversa).

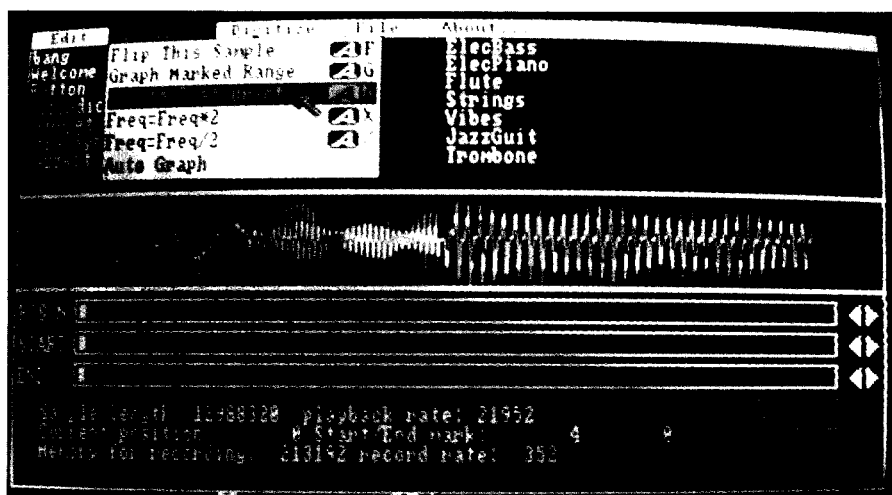
Lo schermo di lavoro subito dopo il caricamento

Per chi, oltre al campionamento, volesse cimentarsi nella generazione di suoni con la tecnica della sintesi, come nei primi sintetizzatori, consigliamo il programma SYNTHIA (della ditta americana The Other Guys) che consente di realizzare suoni per mezzo della sintesi additiva e sottrattiva (è possibile creare suoni tipo l'organo Hammond).

Come usare i suoni della nostra raccolta per fare musica

Una volta realizzati uno o più dischetti di suoni, avrete a disposizione una gamma di strumenti da utilizzare con i programmi di composizione musicale o i sequencer che ormai sono numerosi in commercio.
Citiamo, fra i tanti, il classico Deluxe Music ma anche il recentissimo Music-X, sofisticatissimo programma MIDI che consente di eseguire i brani registrati da una tastiera esterna anche con i suoni interni di Amiga e che permette di suonare in diretta i suoni di Amiga da una tastiera esterna, possibilità questa sempre più diffusa tra i nuovi sequencer MIDI per Amiga.
Questa caratteristica è molto interessante per coloro che, non disponendo ancora di un campionatore musicale, abbiano bisogno delle prestazioni che solo queste macchine offrono.
Con Amiga è possibile sostituire tali macchine e, nonostante alcuni limiti tecnici, dovuti alla ridotta larghezza di banda, non le consentano di pareggiare i campionatori professionali, con un oculato impiego, il dilettante può davvero ottenere risultati sorprendenti.
Volete un esempio?
Provate a "pescare" dal dischetto dei brani Jazz "HOT & COOL JAZZ" per il Deluxe Music il bellissimo suono campionato "LOW PIANO".
Nella parte centrale della tastiera è di una purezza eccezionale.
Bene, caricatelo ora su un sequencer (noi abbiamo provato sul Music Recording Studio della Dr.T's) e collegate una qualunque tastiera MIDI ad Amiga.
Predisponete il programma ad utilizzare la suddetta voce e poi suonatela come solo di piano, magari con un sottofondo di violini da parte del vostro synth.
Provare per credere!

Scheda sul programma Perfect Sound

Questo programma, sviluppato dalla ditta USA "Sunrize", è stato uno dei primissimi campionatori apparsi per Amiga e, tutto sommato, rappresenta ancora oggi un buono standard.
Nella raccolta dei Fish Disk, nel dischetto 50, è presente una copia demo versione 2.0, sviluppata da Anthony Wood.
Nella videata README, viene spiegato che il programma può essere utilizzato per editare suoni digitalizzati oppure con l'hardware Perfect Sound Stereo Digitalizer per registrare ed editare i vostri suoni preferiti. Il prezzo indicato è di 79.95 $ e l'indirizzo è: Sunrize Industries, PO BOX 1453 College Station, TX 77841.
Per la cifra sopra citata, viene proposto l'invio di: un digitalizzatore

Lo schermo di lavoro dopo aver fatto il "pieno" di suoni

che registra in tempo reale stereofonicamente, la più recente versione del programma Perfect Sound, il manuale e una libreria di suoni preregistrati (battiti di mano, aerei, urla, spari ecc..).
Viene, inoltre, chiarito che questa copia del programma (tipico dei Fish Disk) è distribuita amichevolmente. Se vi piace e lo ritenete utile, viene suggerita una donazione di 20 $. In cambio di questo contributo, la Sunrize promette l'invio di: manuale d'uso, l'ultima versione del programma e una copia di "The Captain's Greatest Hits, Vol I", un disco pieno di suoni digitalizzati, in formato IFF, da usare con programmi musicali.

Oggi, questo programma è stato ulteriormente aggiornato e compare in pubblicità nella versione 3.0; tuttavia, la versione Fish si dimostra eccellente per i nostri scopi.
Anche se non disponete del manuale, il funzionamento è molto intuitivo e, dopo un po' di prove, familiarizzerete facilmente con le numerose opzioni. L'azione si svolge in un'unica schermata a sfondo nero, divisa in quattro parti: in quella superiore compaiono i nomi dei file che vengono o caricati col programma o creati successivamente.
Cliccando sul nome si manda in esecuzione il suono relativo.
Nella sezione immediatamente sotto, viene rappresentato il grafico audio del segnale, che può comparire intero o in parte, per studiarne il dettaglio, in funzione della posizione dei tre gadget a cursore contenuti nella sezione sottostante.
L'ultima sezione contiene le informazioni numeriche dei parametri del suono, nonché due gadget che consentono di attivare il suono per intero o per la parte (range) selezionata.
Per andare a "pesca di suoni" sui dischetti dei giochi, è sufficiente caricare Perfect Sound e poi, una volta inserito il dischetto da cui prendere i file IFF, dare il comando LOAD.
Cercando nel menu del dischetto con il gioco in questione, troverete, prima o poi, tali file da "pescare".
Non temete di sbagliare, poiché, se date il Load a qualcosa di diverso, il programma vi dice a video che non si tratta di un file IFF.
Una volta caricato il suono, date il SAVE AS IFF e lo riponete in un dischetto di file di voci che rappresenta il vostro archivio. ▲

Superbase Professional 3.0

Steven Anzovin

Al di fuori del creativo mondo Amiga, in cui trovate grafica, animazioni, video, musica e giochi stupefacenti, si distendono senza fine le righe grigie e ocra dei PC, che macinano, fra le altre cose, immani programmi per la gestione di database. Nella categoria informatica dei lavori "seri", la gestione dei database regna sovrana e i PC compatibili fanno i "galletti" con programmi come dBase III della Ashton-Tate e Paradox della Borland. Amiga può competere in questa arena informatica di "macho"? La risposta è Sì, con Superbase Professional della Precision Software, un sistema per la gestione di database (DBMS) completamente relazionale e di massima qualità, che è insieme potente e facile da usare.

Superbase è un software basato su moduli dalla potenza crescente. Superbase Personal costituisce il livello di ingresso, con gestione di base dei file e dei record, tre modi di rappresentare i dati e capacità di Query e di Report. Superbase Personal 2 estende Superbase Personal fino ad includere un text editor e *funzioni di telecomunicazione*.

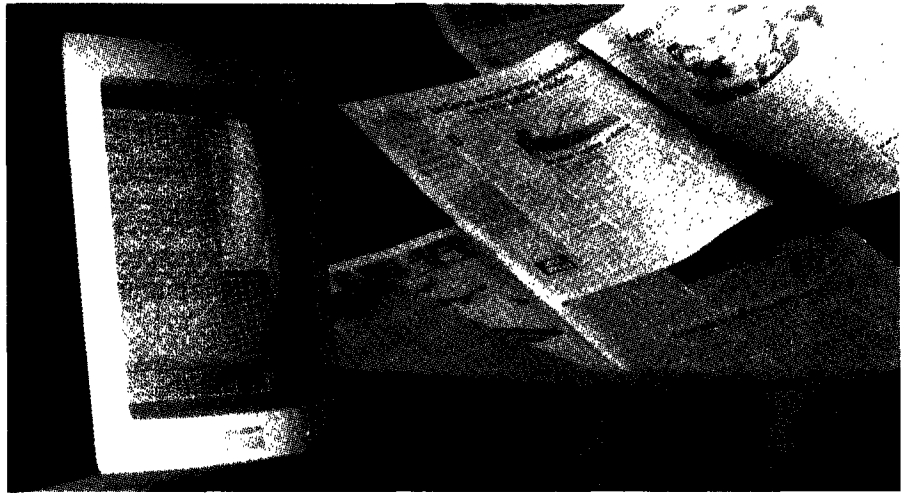
Superbase Professional, qui recensito, aggiunge un Forms Editor, il Database Management Language (linguaggio per la gestione di database) e la compatibilità con ARexx. Ciascuna versione può essere "aggiornata" (up-grade) con la successiva, pagando la differenza di prezzo, rendendo così facile iniziare con il sistema Superbase, per espanderlo man mano che le necessità e le competenze si accrescono. Ciò è inusuale nel mercato dei DBMS, non c'è alcuna versione "home" di dBase III, per esempio, ed è improbabile che ce ne sia mai una. Di Superbase esistono anche versioni per PC e per Atari ST che possono girare sotto la shell GEM OS; i loro file sono compatibili con la versione per Amiga.

Superbase Professional è attualmente una collezione di programmi, basata sul gestore di database, che include un editor di testi, un Forms Editor e un linguaggio per comandi simile al BASIC. Il gestore di database può essere completamente guidato con mouse, menu e requester, ed è dunque molto facile imparare ad usarlo. Per muoversi attraverso un file da un record all'altro, potete usare i controlli posti sul fondo dello schermo, costituiti da pulsanti simili a quelli dei videoregistratori, che permettono di spostarsi in avanti, indietro, in avanti velocemente e così via. I record possono essere visualizzati in tre modi: uno alla volta, con un campo per linea (Record View), un *record* alla volta con campi decisi dall'utente (Form View), o tutti i record insieme in formato tabella (Table View). La barra di scorrimento permette di visualizzare record che sono troppo grandi per stare in uno schermo.

L'opzione di ricerca di Superbase vi permette *di eseguire operazioni* (chiamate key lookups) con molti parametri e variabili. Potete anche usare filtri basati su pattern matching e operatori relazionali, booleani e matematici per visualizzare solo i record i cui campi soddisfano certi criteri, come quelli che contengono un nome specifico o una gamma di valori. Filtri e ricerche sono completamente definibili dal-

l'utente per mezzo di requester ben progettati. Il contenuto dei record può essere modificato in qualsiasi momento e salvato tutto insieme su disco (per diminuire il numero di salvataggi verso il lento floppy disk). Quando create un nuovo database, vi verrà richiesto il nome del file, una password opzionale e la definizione dei dati per ciascun campo del file. Superbase Professional può gestire quasi tutti i tipi di dati trattati da Amiga. Il programma ha i suoi tipi di dati interni: stringa, numero, data/tempo; ma può accedere anche a molti generi di file esterni, fra cui testi in formato ASCII, immagini in formato IFF (per librerie di immagini, uno degli usi migliori per Superbase Professional), suoni in formato IFF e suoni campionati. Inoltre, potete importare file nel formato di dBase III o di altri programmi, come i file .WKS di Lotus. E' possibile stabilire un'ampia gamma di altri criteri per i campi di dati: se i dati siano a sola lettura, se sia necessario calcolarli o convalidarli, se si tratti di costanti. Formule, funzioni e operatori ternari possono essere aggiunti ad ogni campo e verranno eseguiti automaticamente per ogni valore che immetterete. Nel complesso, il numero di possibili varianti per i tipi di dati e di campi fa "vacillare", ma risulta facile da gestire perché le scelte vengono presentate attraverso diversi requester, così potrete prendere in considerazione solo quelle opzioni che vi servono.

Superbase Professional può certamente svolgere lavori "pesanti". Non c'è limite al numero dei campi in un record o al numero di record in un file e la memoria è il solo vincolo sul numero di file che possono essere aperti contemporaneamente. I campi stringa sono limitati a una lunghezza di 255 caratteri, pertanto, per testi più lunghi, vanno usati campi esterni di testo. Il testo per questi campi può essere creato attraverso l'editor built-in di Superbase. Ciascun file deve avere almeno un indice, il massimo è 999 indici.

Le funzioni di Report, il cuore di ogni database, sono gestite dalle funzioni di Query di Superbase Professional. Le Query vengono costruite in maniera molto simile ai filtri, con la possibilità di eseguire ordinamenti veloci su più file. Per decidere il formato del Report, potete usare il Forms Editor, un programma separato per il formato d'impaginazione, che gira in multitasking assieme al gestore di database. Il Forms Editor è utile per progettare modelli elettronici della pagina che volete usare, fatture, ordini, spese e così via, con calcoli built-in, aggiornamenti e relazioni fra dati appartenenti a file diversi.

Il Forms Editor usa un approccio intuitivo, fondato sul mouse, per il disegno della pagina: grazie ad esso la manipolazione dei campi, dei pulsanti e della grafica si effettua muovendo semplicemente l'oggetto in questione nell'esatta posizione. Se avete usato un programma per la definizione del formato di una pagina o un programma di CAD, non dovreste avere problemi nell'imparare ad usare il Forms Editor; è uno degli accessori più belli di Superbase Professional, anche se mi sarebbe piaciuto vedere file di form predefiniti, facilmente personalizzabili. Sono comprese, inoltre, utilità per il mail-merge, la stampa di etichette e le telecomunicazioni.

Per gli utenti avanzati, Superbase Professional include un Database Management Language (linguaggio per la gestione di database) o DML, modellato strettamente (ma non esattamente) sul BASIC. DML vi permette l'accesso a tutti i menu e i comandi di dialogo di Superbase Professional. Potete usare DML per effettuare comandi singoli con esecuzione immediata, come un semplice linguaggio di macro, per automatizzare sequenze ripetitive di comandi e, infine, come un completo strumento di programmazione, che consente di creare menu e requester in un ambiente del tutto personalizzato.

Superbase Professional supporta anche ARexx, il linguaggio sviluppato da William S. Hawes. ARexx, che sta diventando il linguaggio d'interfaccia standard per i programmi di Amiga (altri DBMS, come Microfiche Filer Plus, lo supportano), può essere usato per pilotare ogni programma o dispositivo esterno che risulti compatibile. Per esempio, potreste sviluppare un'applicazione con Superbase Professional in grado di accedere a un database di immagini su videodisco, usando ARexx per pilotare un controller per videodischi compatibile, naturalmente, con ARexx. Per gli sviluppatori, la Precision offre due programmi di supporto avanzato. L'Official Superbase Information Network (OSIN) è un servizio nazionale (su American People/Link) che fornisce supporto tecnico e discussioni sugli attuali sviluppi di Superbase.

Il programma Official Superbase Application Developer (OSAD) della Precision può aiutarvi a progettare nei dettagli applicazioni DBM,

usando Superbase Professional, e può anche assistervi nella commercializzazione del vostro prodotto.
Superbase Professional ha ancora qualche "difetto". Per esempio, la gestione delle directory è inutilmente complessa. Invece di permettervi la selezione di un pulsante o l'inserimento del nome di un path, in un file requester del tipo "Load" o "Save", come avviene nella maggior parte dei programmi per Amiga, dovete eseguire comandi di menu separati per leggere e cambiare drive e directory. Il Text Editor è un'utile aggiunta al programma, ma difficilmente potrà essere considerato un word processor potente. Inoltre, Superbase Professional usa comandi sostitutivi da tastiera che non corrispondono a quelli utilizzati in altri editor di testi per Amiga. Probabilmente, preferirete far girare in multitasking il vostro word processor preferito. La documentazione, sebbene completa e ben scritta, è povera di indici. Mi piacerebbe vedere anche un help in linea, o perlomeno una lista dei comandi da tastiera per il Text Editor. C'è un piccolo bug nascosto in Superbase Professional. Sono riuscito a mandare in crash il programma attraverso il Forms Editor, aprendo un file esterno di testo presente sul disco del Forms Editor e poi tentando di chiudere il file con il gadget di chiusura. Un'altra "seccatura" è la protezione contro le copie, che richiede un dongle nella porta joystick (game port 2) mentre il programma gira. Se perdete il dongle e non avete rispedito il modulo per la registrazione della garanzia, potete dire addio al vostro Superbase. Raccomando vivamente di tenere sempre il dongle inserito o legato al vostro Amiga. Meglio ancora, inviate il vostro modulo di registrazione e poi comprate per 10 dollari la versione sprotetta. Resta da chiedersi perché la Precision non elimini il dongle e la procedura aggiuntiva, non aggiunga 10 dollari al prezzo del programma e rimuova insieme la protezione contro la copia. A parte il dongle, Superbase Professional è il più potente database disponibile per Amiga. Il numero di funzioni disponibili è stupefacente. Può essere adattato a quasi ogni compito di gestione di database, con massima velocità e programmazione minima. Se intendete dedicarvi del tempo, il linguaggio interno DML e la compatibilità con ARexx danno a SuperBase Professional una flessibilità illimitata per la costruzione di applicazioni personalizzate per la gestione di database. In breve, qualsiasi cosa possiate fare con database professionali per PC, potete farla con Superbase Professional. ▲

Pacific Motion

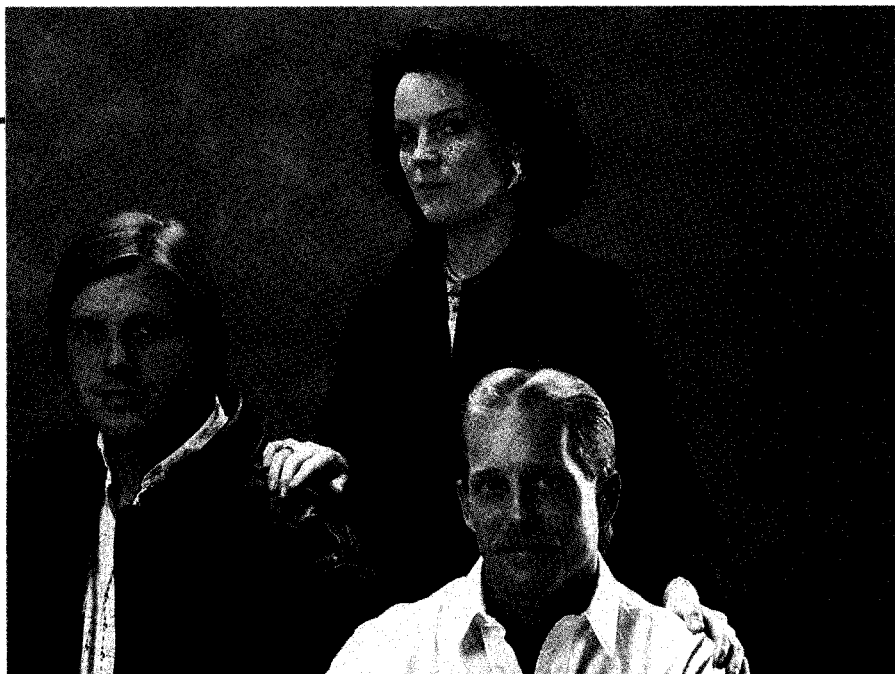
Lo studio di design di Hollywood in cui la grafica di Amiga trasforma le idee in produzioni di alto livello

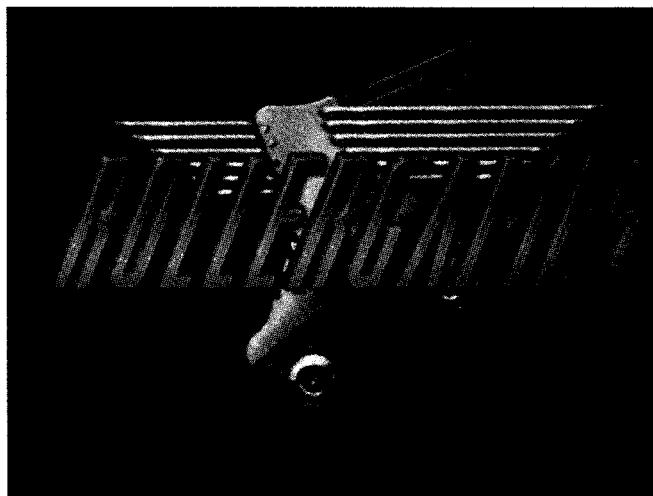
Ben e James Means

E' venerdì notte e ancora una volta siete usciti per andare al cinema a masticare seriamente del popcorn. Vi sedete e, quando le luci si abbassano, bevete un sorso di una bibita. Improvvisamente un magnifico titolo si libra verso l'alto e rotea attraverso il grande schermo e voi siete trascinati entro l'esplosivo mondo pubblicitario dei "prossimamente" hollywoodiani. Non appena il promo finisce, siete catturati ancora dalla realtà alternativa di un altro seducente bagliore di celluloide. C'è sempre lo spettacolo diurno di domani...

Poco importa all'incauto spettatore sapere che questo promo con i suoi titoli animati in maniera ingegnosa, noto come trailer nel mondo dell'industria, potrebbe benissimo essere stato disegnato su Amiga presso la casa di progettazione grafica di Burbank, la Pacific Motion. Rick Probst, Robert Visty e Barbara Eddy disegnano i loro stupefacenti titoli di testa e la loro grafica promozionale con cel animator tradizionali, edit bay per effetti speciali e computer grafica di alto livello elaborata su workstation Bosch, Symbolics e Quantel Paintbox. Ma prima di prenotare un solo minuto del carissimo tempo dello studio, la Pacific Motion elabora il progetto di ogni titolo, grafica ed effetto sulla sua piccola schiera di Amiga.

I migliori clienti hollywoodiani richiedono il massimo, e la Pacific Motion ha usato Amiga per realizzare, con puntualità e secondo i piani finanziari previsti, la grafica di film come Cocktail, Young guns, Can't buy me love, Three men and a baby e Cin-

Pacific MOTION

Pacific Motion (da sinistra a destra) Rick Probst, Barbara Eddy e Robert Visty

Il logo di "Roller Games" generato da Amiga ... e la versione finale ritoccata con Quantel Paintbox

derella. Ancora solo tre anni fa, Rick Probst "sgobbava" per produrre grafica presso la Cimarron Production nel solito modo: con pennelli, pennarelli e aerografo. Le storyboard sono una serie di disegni che mostrano i singoli quadri di una sequenza di animazione; il regista e il produttore possono, in questo modo, farsi un'idea di come apparirà il prodotto grafico finale, suggerire modifiche o approvare la produzione finale. Rick disegnava a casa sul suo Commodore 64, quando comprese per la prima volta che i computer potevano eliminare la fatica dalla noiosa fase di realizzazione delle storyboard. Ma il suo superiore presso la Cimarron non volle investire sulla sua idea, così Probst rassegnò le sue dimissioni, comprò un Amiga 1000 e nel tardo 1986 fondò la Precision Image con il produttore Robert Visty al fine di realizzare grafica cinematografica. Due anni più tardi, si associarono con Barbara Eddy per creare la Pacific Motion e si espansero verso il settore della produzione televisiva. Nell'autunno del 1989 il loro lavoro è apparso nelle produzioni della Disney: "Chip and Dale's Rescue Ranger", "Roller-Games", "Whell of Fortune", "Jeopardy", "Moments to love", e nel film di De Niro e Fonda "Stanley and Iris". Abbiamo parlato con Probst e Visty, chiedendogli in che modo Amiga li abbia aiutati a crescere così rapidamente.

AMIGA MAGAZINE: Voi e Amiga siete partiti insieme. Quale ruolo ha giocato Amiga nell'espansione della vostra azienda di design?

PROBST: Nel novembre del 1986 affittammo un piccolo appartamento di due stanze. Avevo il mio aerografo, un compressore e un Amiga 1000 con soli 512K; iniziammo facendo storyboard con una stampante Okimate 20 e subito le presentammo alla Disney.
Tradizionalmente i grafici realizzano le storyboard con aerografo, pennelli e pennarelli. Il vantaggio offerto dal computer è che puoi usare una parte di ciascun immagine per l'immagine successiva o la precedente. Così quando stai lavorando a un piccolo effetto zoom, puoi cambiare la scala generale e stampare l'immagine come un fotogramma della storyboard, per creare facilmente e rapidamente l'effetto zoom.
Usare l'aerografo è lento e noioso. Se io volessi usare l'aerografo per questo stesso effetto zoom, dovrei ritagliare mascherine in ognuna delle scale della figura e dipingerle con l'aerografo. Poi una volta tolte le mascherine e l'effetto non fosse buono, dovrei rimettere nuovamente insieme le mascherine e ricominciare daccapo. Ci si sporca, si impiega tutta la notte e si ottiene un'unica storyboard.
Su Amiga, posso fare tre o quattro storyboard diverse nello stesso tempo che impiego a farne una con i metodi tradizionali. Se un altro disegnatore arriva con una sola storyboard e tu arrivi con molte idee, allora hai già ottenuto un certo vantaggio. Per questo sin dall'inizio usai Amiga, perché stavo facendo molte storyboard. Avevo scoperto che Amiga era il modo migliore per farle.

AMIGA MAGAZINE: Amiga ti ha dato un vantaggio in questo competitivo settore industriale?

PROBST: Sì... con cicli produttivi più veloci. In questa città il tempo di produzione è tutto. Arrivano alla porta il lunedì e vogliono che i titoli del film siano pronti per venerdì! E' dura, ma può essere fatto. Spesso il cliente ha bisogno di consegnare il prodotto al network, per rispettare certe scansioni temporali, o alle sale, per la distribuzione di un nuovo film. Così si ha a che fare con qualcosa di inesorabile, cui bisogna far fronte. Ero solito chiedere tre giorni per consegnare una storyboard realizzata con l'aerografo e speravo che avessimo già una idea pronta per il disegno. Con Amiga, posso normalmente mostrarla il giorno successivo; così Amiga permette di risparmiare due giorni consentendo cicli produttivi più rapidi.

VISTY: Per vendere un progetto a un produttore, devi visualizzarlo per lui, prima che metta un dollaro sul tavolo e dica "Andate avanti". Il vantag-

gio di Amiga, rispetto ai metodi tradizionali, è che posso concretizzare un'idea molto più velocemente, per portarla di fronte a qualcuno. La gente vede ciò che sta comprando, approva i progetti più velocemente, i cambiamenti diventano più rapidi. Più progetti giungono alla fase di completamento.
Alcuni dei nostri clienti sono veramente specializzati in cambi dell'ultimo minuto, così Amiga ci ha aiutati non poco anche in questi casi. Un produttore può essere in Canada, mentre il suo regista può essere a New York. Con questo sistema, non ho problemi nel fare copie multiple di uno stampato in pochi minuti e spedirlo di notte, dovunque siano. Poi possono sollevare il telefono mentre tutti guardano la stessa cosa. Se stai lavorando con un illustratore, devi mandare fuori il materiale perché venga fotografato, sviluppare i negativi, fare più stampe; perdi almeno un giorno. Amiga "rimuove" questi tempi morti dalla produzione e aiuta a condurre avanti un progetto in maniera spedita.
Come produttore, amo Amiga perché quando pianifico finanziariamente un progetto, conosco tutti i miei parametri in anticipo. Così non ho problemi nel calcolare il costo di qualcosa, e non ho problemi nel realizzarla secondo i tempi previsti, cosa che contribuisce per il 50% alla soddisfazione del cliente.
AMIGA MAGAZINE: Perché lavorate professionalmente su Amiga invece che su qualche sistema di livello più alto?
VISTY: Parte della nostra strategia come casa di design è quella di rimanere tali; se possedessi equipaggiamenti di alto livello e di alto costo, come una stazione di animazione o un edit bay, allora dovrei noleggiare quelle macchine e darmi da fare per pagarle: non sarei più una designer. Inoltre, sarei costretto a lavorare sulla macchina che ho comprato. Come designer, voglio lavorare su qualunque cosa sia uscita, ogni volta che lo desidero, mentre molte case di postproduzione progettano sempre per le macchine che posseggono. Amiga ci dà una grande flessibilità nel design, che ad altri sarà preclusa finché resteranno legati a un solo tipo di macchina.
AMIGA MAGAZINE: Deluxe Paint III ha causato qualche cambiamento significativo nel vostro lavoro?
VISTY: Amiga ci ha permesso di fare una storyboard composta ancora da singoli fotogrammi, ma Deluxe Paint III ci ha permesso di fare ciò che chiamiamo temp animation, che è una storyboard animata su video. Nessun altro lo sta facendo a livello professionale. Queste storyboard animate e registrate su nastro hanno costituito il passo che ci ha dato un vantaggio sulla concorrenza. La nostra temp animation assomiglia molto al prodotto finale. Le tecniche classiche permettono un livello di somiglianza meno elevato e il cliente deve interpretare una scena disegnata immaginando come apparirà alla fine sul nastro magnetico. Ciò che Amiga mostra loro è molto vicino a ciò che verrà trasmesso, così le sorprese sono ridotte.
PROBST: Alcune delle persone cui mostriamo i progetti, sono sufficientemente esperte da comprendere una stoyboard, quando gliela mostriamo. Ma altre persone incontrano delle difficoltà nel visualizzare ciò che accade tra i singoli fotogrammi. Quando si mette una storyboard animata su nastro, ci sono meno spiegazioni da dare. Per questo Deluxe Paint III è grande. Invece di avere una panoramica grafica attraverso tre quadri nello stampato della storyboard, la panoramica avviene con più dozzine di fotogrammi, su video.
VISTY: Risparmiamo anche molto sui costi. Abbiamo fatto animazioni su Amiga usando Deluxe Paint III, le abbiamo stampate e poi prese e utilizzate come una base per animazioni realizzate con cel animator standard.
Normalmente dovresti pagare qualcuno per il tempo che impiega nel disegnare i diversi quadri. Poi dovresti riprendere l'animazione con una telecamera, controllarla, modificarla, e, infine, controllare i cambiamenti. Ora non ho più bisogno di fare tutto ciò: la prima stampa che ottengo da Amiga è già compiuta. Se sei in fase produttiva, hai risparmiato una settimana e mezza e lavoro grafico per molte migliaia di dollari. Animatori e artisti che lavorano su Paintbox costano; per un progetto affidato a due di queste persone spenderei tanto quanto mi costerebbe un Amiga. Dunque, preferisci che quei dollari finiscano nelle tue tasche o nelle tasche di un altro?
Puoi anche riversare direttamente l'uscita RGB [di Amiga] nel Paintbox e poi rifinire il disegno, invece di ridisegnarlo completamente. Con i costi attuali, puoi risparmiare 300 o 400 dollari ogni due o tre ore, in quanto non si è costretti a ridisegnare da zero la propria grafica.
AMIGA MAGAZINE: Che cosa pensano i vostri clienti di Deluxe Paint III?
PROBST: L'ho usato per "Rollergames" ed è stato utile per assicurarsi l'affare. "Rollergames" è una evoluzione di Roller Derby che viene trasmesso in concorrenza con "Saturday Night Live". Usa il modello di Roller Derby, in cui un pattinatore, detto jetter, fa una fuga abbandonando il gruppo e fa punti per ogni pattinatore che riesce a superare. Abbiamo fatto tutta la preproduzione per la grafica di apertura, il logo della squadra e le inquadrature prestabilite di uno stadio su Amiga. Così tutto quello che i disegnatori del Paintbox hanno dovuto fare, è stato rifinire alcuni particolari e migliorare i colori.
Ho creato lo sfondo per il cartellone elettronico dei punti di "Rollergames" con il Paintbox, ma ho usato Amiga per sovrapporre le immagini che cambiano ad ogni show settimanale, come i punti o gli indicatori del tempo. Jeff Bruette scrisse un programma specifico per Amiga per aggiornare l'orologio e il punteggio durante il gioco. Abbiamo ancora un cartellone elettronico con l'aspetto che il Paintbox è in grado di generare, ma Amiga ci risparmia il costo di un Paintbox ogni settimana.
AMIGA MAGAZINE: Che cosa usate come dispositivi di input?
PROBST: I miei tre dispositivi di input

sono il mouse, il digitalizzatore ed Easyl. Ho abbozzato il cartellone elettronico per "Rollergames" prima su carta, per assicurarmi che ogni cosa funzionasse bene; poi ho riprodotto il disegno con una telecamera, l'ho digitalizzato, ne ho ridotto i colori con Pixmate e ho cominciato a disegnare su di esso con il mouse. Recentemente ho lavorato su alcune immagini di Chip e Dale che ci ha inviato la Disney. Le ho distese sulla Easyl e ricalcate con un pennello dello spessore di un pixel; l'input di Chip e Dale è stato veramente buono, migliore di quanto avrei potuto ottenere con la digitalizzazione.
Per il disegno a mano libera, uso la Easyl; se voglio caricare un'immagine di tipo fotografico uso Digi-View. Oppure posso iniziare da una font. Ho comprato Masterpiece Font Collection che è favoloso per ciò che faccio, perché sono tutti font grandi, alti 2.5" in alta risoluzione, che è esattamente ciò di cui ho bisogno per il design dei titoli e la loro elaborazione.

AMIGA MAGAZINE: Quale altro software e hardware usate?

PROBST: Ho appena ricevuto una nuova versione di Digi Works 3D, che mi permette di tracciare linee attorno al mio carattere. Ora posso trasformare le immagini bitmap dei caratteri e convertirle in font tridimensionali. E' un favoloso software. Amo anche Elan Performer. Quando sono stato a Detroit per la conferenza Broadcast Promotional Marketing Executives (BPME), avevo trasformato quasi un'intera tastiera in hot-key. I primi sei tasti funzioni lanciavano delle animazioni, e il resto della tastiera era costituito da singole immagini. Potevo entrare in un'animazione premendo semplicemente un tasto, oppure, se volevo fermarmi e parlare, mi bastava premere la barra spaziatrice e iniziava un ciclo che passava attraverso tutte le animazioni e le singole immagini. E' un grande programma per realizzare dimostrativi.
La macchina che avevo a Detroit montava la scheda GVP 030, 4 Mega di RAM a 32 bit e due schede da 2 Mega. Con 9 Mega sulla macchina, potevo caricare le animazioni e anche le immagini per visualizzarle senza interruzioni tutto il giorno, mentre Elan Performer non ha mai dovuto fare un accesso al disco durante la performance. Dunque, più memoria avrai, meglio sarà con quel programma.

AMIGA MAGAZINE: Che cosa rende Amiga così adatto al design?

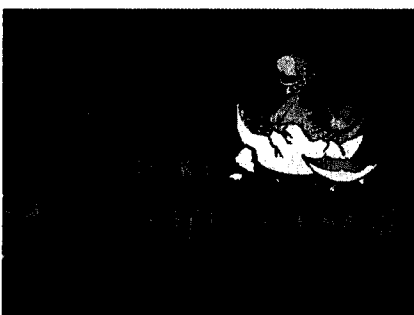
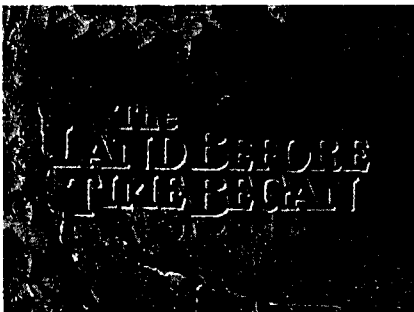
PROBST: Il vantaggio di Amiga è che si possono trasferire le tue idee dal cervello allo schermo molto rapidamente, prima che sia sopraggiunta una nuova idea. Non si perdono idee a causa di pennarelli, aerografi o tecnologia elaborata. Quando disegno, inizio elaborando dei caratteri. Gli aggiungo un profilo, oppure una superficie particolare e poi la elaboro come stencil in Deluxe Paint. Dopo di che, salvato il risultato dell'idea precedente, inizio con una nuova idea e riprendo il disegno iniziale, tentando una nuova via. Amiga è interattivo proprio a questo modo; lavora con te. Se dovete aspettare per certe cose, cancella l'attesa.

AMIGA MAGAZINE: Quale effetto avrà complessivamente Amiga in questo settore industriale?

PROBST: Con Amiga ad un così basso livello di costi sul mercato, ogni televisione via cavo può fare con esso qualsiasi tipo di cosa. La nostra televisione locale via cavo usa una grafica terribile che sembra fatta con qualche programma di disegno per IBM di basso costo. Linee super-dentellate, colori veramente brutti; puoi vederne i limiti immediatamente. Amiga raggiungerà il massimo potenziale di crescita con l'uscita del supporto per UNIX e X-Windows. Il disegnatore dell'ufficio vicino ha una workstation Iris su cui gira del software specifico chiamato Prism. Io potrei creare file di oggetti tridimensionali con Sculpt 4D, collegare una Ethernet, mandare quegli oggetti al VAX ed elaborarli in circa un minuto per fotogramma. Una volta installato il collegamento, la possibilità di fare design usando Amiga come terminale per workstation diventa una realtà.

AMIGA MAGAZINE: Che cosa ha fatto Amiga per la vostra impresa?

PROBST: Amiga è stato lo strumento definitivo per il design. Fa qualsiasi cosa; quando voglio lavorare su un'idea, posso vedere abbastanza bene come apparirà. Posso fare cose che proprio non potevo fare in precedenza. E' una grande interfaccia per prendere decisioni, e penso che sta diventando difficile superarla. ▲

Alcune schermate di titoli generate da Amiga

Microsoft Press ha scelto Jackson per tradurre in italiano la bibbia dell'MS-DOS

**TUTTI I SEGRETI DELL'MS-DOS
IL SOFTWARE PIU' DIFFUSO NEL MONDO**

Il punto di riferimento che non verrà scalfito neppure con l'avvento di nuove release.
Lo sviluppo storico dell'MS-DOS, la gestione dei comandi, delle directory e della memoria, le utility e il debugging.
Lo strumento insostituibile per i programmatori di qualsiasi livello, ma anche per apprendere o perfezionare le proprie conoscenze.

DA UNO STAFF DI SUPER ESPERTI TUTTO IL KNOW-HOW DELLA MICROSOFT PRESS

Scritto dai migliori programmatori del mondo Microsoft, tutti consulenti informatici e professionisti che hanno dedicato la loro vita alle problematiche che gravitano attorno all'MS-DOS, coordinati da Ray Duncan, con la supervisione di Bill Gates, il genio del software, fondatore e attuale Chairman di Microsoft Corporation.

UNA PIETRA MILIARE NELLA STORIA DEL COMPUTER IN UN'OPERA UNICA

•14 autori •65 consulenti tecnici •1.600 pagine suddivise in 7 sezioni: lo sviluppo dell'MS-DOS - La programmazione in ambiente MS-DOS - I comandi dell'utente - Le utility di programmazione - Le chiamate del sistema - 100 pagine di appendici per codici degli errori, set di caratteri e set di istruzioni - Un indice dei comandi e un indice delle chiamate di sistema.

1.600 PAGINE
L. 190.000

I libri del Gruppo Editoriale Jackson sono in vendita presso le migliori librerie e computershop. Se ti è più comodo acquistarli per corrispondenza utilizza questo coupon.

Per ordinare il libro **Enciclopedia dell'MS-DOS** utilizzate questa cedola. Ritagliate e spedite in busta chiusa a: **GRUPPO EDITORIALE JACKSON** - Via Rosellini 12 - 20124 Milano

Titolo	Codice	Prezzo	Quantità	Totale lire
Enciclopedia dell' MS - DOS	R881	L. 190.000		

Aggiungere all'ordine L. 4.500 come contributo fisso di spese postali.

❑ Sono titolare della Jackson Card'90 n°: ______ e ho diritto allo sconto del 10% (fino al 31/12/90)

❑ Non sono titolare

MODALITÀ DI PAGAMENTO: ❑ Contro Assegno postale al ricevimento dei volumi

❑ Assegno bancario n° ________ di L. ________ Banca ________

❑ *Ho effettuato il pagamento a mezzo:* ❑ *Versamento sul c/c post. n° 11666203 a Voi intestato* e allego fotocopia della ricevuta

❑ Addebitatemi l'importo di L. ________ sulla carta di credito: ❑ Visa ❑ American Express ❑ Diners Club ❑ Carta Si
n° ________ data di scadenza ________

❑ Richiedo fattura (Partita IVA n° ________)

Cognome e Nome ________

Via ________ n° ________

Cap ________ Città ________ Prov. ________

Tel. ________ Data ________ Firma ________

M

Un programma musicale semplice da usare ma molto potente

David English

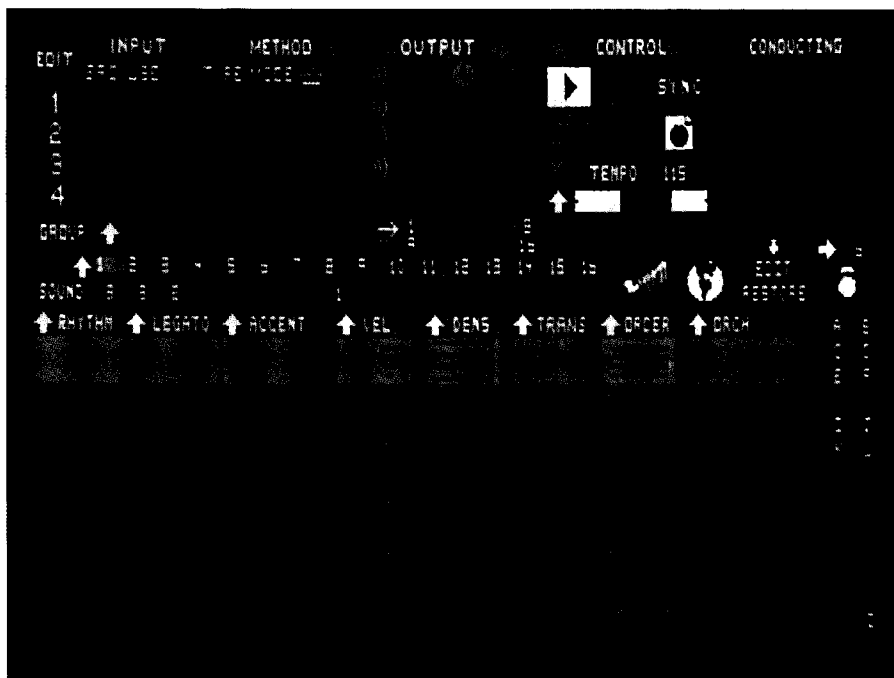
Chiudete i vostri occhi per un momento. Immaginate un programma musicale facile da usare, che vi lasci passare rapidamente attraverso centinaia di combinazioni musicali, premendo solo il tasto del mouse. Immaginate che, con questo programma, il vostro computer assuma un ruolo intelligente nelle vostre decisioni musicali. Immaginate un programma che non richieda particolare talento musicale per pilotarlo, solo un buon orecchio per riconoscere quando qualcosa suona in maniera interessante. Ora immaginate che questo programma, facile da usare, crei composizioni musicali che suonano come quelle che si sentono alla radio, così bene che molti compositori professionisti le utilizzano per il loro lavoro, inclusi Jan Hammer, Laurie Anderson, Wendy Carlos, Stanley Clarke, Mark Styles, Larry Fast, Steve Roach, Laurie Spiegel, Pete Townshend, Michael Hoenig e Grover Jackson. Infine, immaginate di poter far girare questo miracolo moderno su un Amiga con un Megabyte, e che, per usarlo, non abbiate bisogno di una tastiera MIDI (ma potete farne uso, se l'avete). OK, ora potete aprire gli occhi. Benvenuti in M. M è un programma insolito.

Quando fu rilasciato nel 1987 per il Macintosh, il Computer Music Journal del MIT lo nominò il primo programma per personal computer in grado di effettuare composizioni mediante algoritmi di potenza professionale (un programma di composizione mediante algoritmi è un programma per computer che compone musica con l'aiuto del suo operatore umano). Da una parte è un programma capace di esplorazioni musicali serie, persino scientifiche. Dall'altra parte, M è un programma che può divertire un bambino di 4 anni. Un bambino può muovere il mouse, selezionare i diversi riquadri, e avvertire immediatamente cambiamenti distinti nella musica. Dopo un po' di tempo, un bambino (o il bambino dentro ognuno di noi) potrebbe imparare a comporre musica e sviluppare un certo orecchio per le strutture musicali. Ho usato M su Macintosh sin dal tardo 1987, appena comparve. Scaricai una versione dimostrativa da PAN (Performing Art Network) e immediatamente me ne innamorai. Il demo stuzzicò solamente il mio appetito. Pagai con gioia per avere la versione completa. Quando Intelligent Music portò il programma su Amiga, ero ansioso di vedere come il mio vecchio amico si sarebbe trovato nel nuovo ambiente.

Felicemente! La versione per Amiga ha quasi tutte le caratteristiche di quella per Macintosh (per qualche ragione non ha il Time Distortion). Secondo la Intelligent Music, può persino leggere file M per Macintosh (che converte internamente) e file nello standard MIDI per Amiga. La versione per Amiga aggiunge sia il multitasking (il MultiFinder per il Macintosh si sta, come dire, ancora evolvendo), che suoni interni campionati (le versioni per Macintosh e PC lavorano solo con l'interfaccia MIDI). I suoni interni aprono il programma a un pubblico ben più vasto dei soli possessori di MIDI.

Sebbene M possa all'inizio sembrare complicato, dopo un po' di pratica, dovreste trovarlo facile da usare. Dapprima, si inseriscono le note in ciascuno dei quattro canali (o

Player) indipendenti, o con una tastiera MIDI, o utilizzando il mouse per selezionare la tastiera sullo schermo. A questo punto M mostra realmente la sua stoffa. Vi permette di alterare le vostre note con otto tipi di variabili: Accent (loud/soft), Legato (long/short), Rhythm (now/later), Tempo (fast/slow), Transposition (consonant/dissonant), Note Order (original/random), Velocity (hard/weak), e Density (many/few). Potete anche controllare la combinazione di strumenti o suoni che eseguono le note. In breve, M vi permette di variare la vostra sequenza di note quasi in modo immaginabile.

Fortunatamente, M rende molto facile tentare diverse combinazioni musicali per trovare quella che vi piace. La maggior parte delle variabili sono rappresentate sullo schermo con una colonna o riga di sei riquadri (chiamate Variable Position). Basta selezionare una volta un riquadro, perché quella variabile influenzi immediatamente le vostre note. Lo si selezioni due volte, e otterrete una finestra di editing, che vi permetterà di riconfigurare la variabile. Per esempio, scegliete uno dei riquadri relativi a Note Order e selezionatelo due volte per far apparire la sua finestra di editing. Vedrete quattro barre orizzontali, una per ogni Player. Ciascuna barra rappresenta 100 unità o il 100% dell'ordinamento delle note. Usando il mouse per far slittare i tre numeri verso la sinistra, potrete dividere la barra in tre parti che indicano tipi diversi di ordinamento delle note: Original Order (in cui le vostre note vengono suonate esattamente come sono state caricate), Cyclic Random (in cui le vostre note originali vengono riordinate in maniera casuale, ma vengono suonate ripetendo sempre lo stesso ordine) e Utterly Random (in cui ciascuna nota è arbitrariamente scelta fra le vostre note originali e l'ordine non viene mai ripetuto). Potreste stabilire il 42% di Original Order, il 37% di Cyclic Random e il 21% di Utterly Random, oppure qualsiasi altra combinazione che raggiunga il totale di 100%.

Questo può essere fatto per ciascuno dei quattro Player della finestra del Note Order e per ciascuno degli altri cinque riquadri relativi a Note Order, stabilendo per ciascuno di essi un diverso set di valori. M rende questo processo intuitivo. Potete suonare le vostre note mentre aggiornate ciascun riquadro, capendo immediamente come i diversi parametri influiscano sulla musica (stiamo parlando di milioni di possibilità fra tutte le diverse combinazioni, così è ragionevole usare l'orecchio per ridurre le possibilità, al fine di giungere a un ambiente di lavoro più selezionato).

Se desideraste essere più metodici nella selezione dei riquadri, provate la Conducting Grid, un'area dello schermo costituita da un insieme di 6x6 riquadri. Ogni variabile ha una Conducting Arrow, che può essere abilitata o disabilitata e modificata facendola puntare verso l'alto, il basso, la sinistra o la destra.

Non appena muovete il mouse sulla griglia, il cursore diventa una bacchetta da direttore d'orchestra. Muovendo la bacchetta attraverso la Conducting Grid, potrete modificare in maniera sistematica le diverse variabili con un singolo movimento, dirigendo ogni variabile nel rispetto della direzione della sua Conducting Arrow.

Quando avete trovato una combinazione di variabili che vi piace in maniera particolare, potrete effettuare uno Snapshot dei controlli presenti sullo schermo. Con gli Snapshot, potete salvare e ricaricare fino a 26 diverse combinazioni di variabili. Dapprima selezionate il pulsante Blink Everything. Poi selezionate uno dei piccoli riquadri vuoti al di sotto dell'immagine della macchina fotografica, ed esso verrà riempito con una lettera per ricordarvi che contiene uno Snapshot.

Non avete bisogno di una tastiera MIDI per usare M. Il pacchetto include un disco di strumenti con 23 suoni campionati in formato IFF-8SVX. Questi includono 12 percussioni, 4 effetti (CowMoo, Bark, Voice e BangZoom), 3 strumenti a corda, 2 a fiato e 2 suoni di sintetizzatore, assieme ad alcune musiche dimostrative che mettono in mostra le potenzialità di M. Il file dimostrativo Bonk è particolarmente ben fatto, con la sua intricata tessitura di percussioni. Caricate i file dimostrativi; dovete ascoltare il programma al lavoro per rendervi conto di quanto sia potente. Potete ordinare un di-

sco dimostrativo di M per 10 dollari presso la Intelligent Music. Il programma dimostrativo possiede tutte le funzioni di quello regolare tranne per il fatto che non potete salvare il lavoro e dovete effettuare un reset dopo 10-40 minuti. Ha sei suoni campionati e due file dimostrativi.
La documentazione di M è eccellente. Il tutorial dovrebbe permettervi di usare il programma dopo pochissimo tempo. Riceverete anche un'iscrizione gratuita a PAN (normalmente 150 dollari) e una copia di RSG (un programma shareware che vi permetterà di editare file IFF-8SVX). Il programma è protetto: si richiedono, all'avvio del programma, delle parole chiave tratte dal manuale. Potete ricevere una versione sprotetta inviando un modulo firmato a Intelligent Music assieme a 50 dollari. Se usate M per comporre musica con tastiere MIDI, vi accorgerete che è particolarmente adatto a gestire suoni tipo percussioni, che non si prolungano a lungo. Siccome eccelle nel manipolare le note, dovreste scegliere voci del sintetizzatore che vi permettono di distinguere una nota dall'altra. Ho usato M per creare intricate sequenze ritmiche e poi le ho importate in un programma sequencer (attraverso file MIDI). Di là ho aggiunto altre tracce MIDI per creare un brano completo. Secondo me, M costituisce un modo notevole per iniziare, specie quando è necessario allontanarsi dalle normali vie musicali. Questo programma mi piace veramente. E' divertente. E' difficile non ridere quando si ascolta un coro di voci, una che dice "bang-zoom" e un'altra che emette un "moooo" di felicità, tutto perfettamente sincopato. Dopo un'ora o due con M, sarete convinti d'aver creato un capolavoro. Perché opporsi? E' più divertente di ogni gioco che abbia mai visto (incluso Tetris) e più efficace di qualsiasi altro programma musicale che potreste probabilmente trovare. ▲

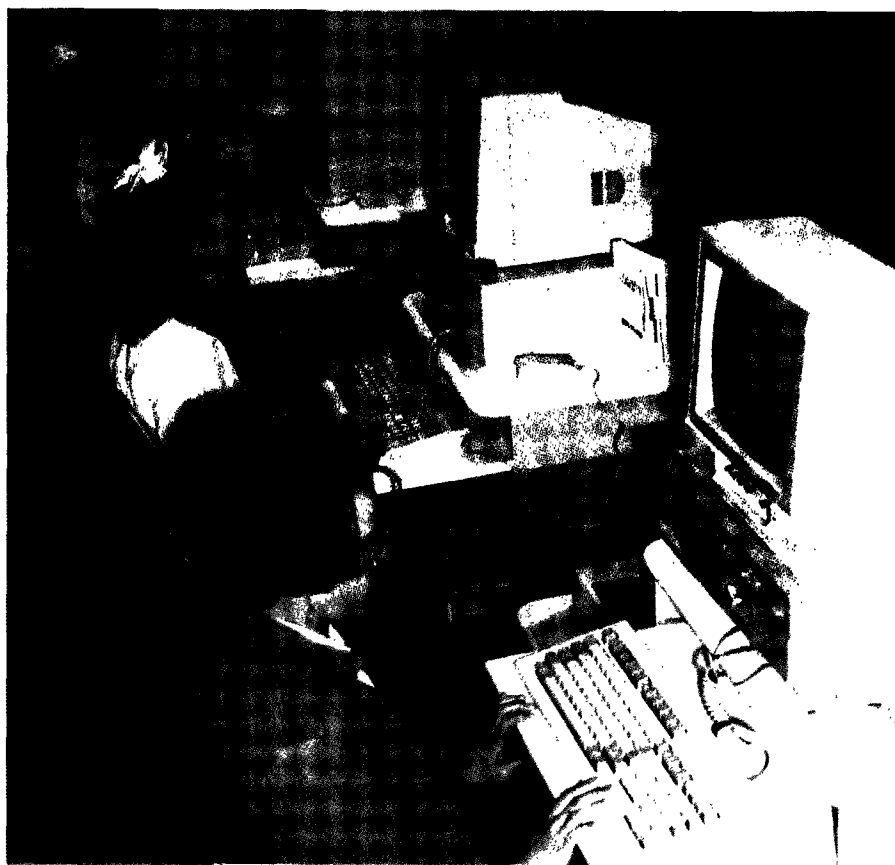
Amiga per il progetto Columbus

A Pregnana Milanese prende il via un laboratorio di informatica multimediale per la didattica della scuola dell'obbligo. Amiga 2000 Education ne rappresenta "il cuore"

Luisella Acquati

Si chiama progetto Columbus, alla scoperta di nuovi mondi, la struttura didattica improntata alle nuove tecnologie messa a punto da docenti e presidenza della scuola media statale "A.Rizzoli" di Pregnana Milanese, in collaborazione con l'Amministrazione comunale che ha voluto l'iniziativa e ai tecnici che l'hanno realizzata.
Si tratta di un laboratorio di informatica multimediale inaugurato il 20 Aprile scorso che ha come obiettivo l'avvicinamento dei ragazzi all'uso del computer per una diffusione della "informatizzazione della cultura" e favorire il collegamento tra scuola e mondo del lavoro.
Questa esigenza si sviluppa maggiormente nella scuola dell'obbligo, luogo privilegiato del processo innovativo che sta caratterizzando la nostra società. In mancanza di una precisa regolamentazione nazionale, però, le scuole più avanzate promuovono progetti pilota studiati dai vari Enti locali in forma singola o consortile grazie alla collaborazione di docenti e aziende.
In questa ottica è nato Columbus studiato da una apposita commissione informatica che ha stilato le basi del progetto sostenuta, in questo, anche dalla società di servizi informatici "Risorse per la didattica".
La configurazione hardware del laboratorio è costituita da 11 postazioni di lavoro, di cui una a disposizione dell'insegnante, composte da un computer Amiga 2000 compatibile AT ed equipaggiato di monitor a colori, disco fisso, disk drive e stampante a colori. Il software comprende programmi MS- DOS per scrittu-

ra testi (word processor), preparazione banche dati (data base), applicazioni grafiche musicali, preparazione di tabelle di calcolo, preparazione di statistiche.
Amiga 2000 Education affianca lo standard MS-DOS con una scheda IBM compatibile a notevoli capacità grafiche.
Lo strumento consente la contemporaneità d'uso di più programmi e pertanto le potenzialità di utilizzo dei software Amiga e MS-DOS sono valorizzate dall'hardware a disposizione. La flessibilità dello strumento, quindi, si rivela particolarmente interessante per l'utilizzo nei programmi della scuola italiana.
Sono previsti programmi di formazione per i docenti, una assistenza telefonica su ogni programma e in futuro si pensa di poter realizzare un collegamento in rete fra le varie workstation.
I programmi di lavoro permetteranno ad ogni studente di esercitarsi almeno due ore alla settimana per favorire l'applicazione e l'uso del computer in tutte le materie curriculari. A questo scopo l'aula laboratorio è stata strutturata per consentire un rapporto ottimale macchina-allievo (una macchina ogni due allievi). Il laboratorio è di proprietà dell'Amministrazione comunale che lo ha affidato in uso alla scuola Rizzoli e si pone come obiettivo di costruire, da un lato, le conoscenze disciplinari e dall'altro le competenze professionali.
Il progetto prevede l'uso del computer per il trattamento dei testi, l'uso e la creazione di data-base, l'impiego di fogli elettronici e l'utilizzo di pacchetti grafici.
Per le diverse discipline si prevedono numerose attività.
Per Italiano, per esempio, si parla dell'utilizzazione di un programma di video-scrittura per il trattamento testi al fine di: dividere un testo in sequenze, compilare un riassunto, identificare i personaggi, comporre un tema; formare un archivio specifico dei lavori svolti e avviarne la consultazione. Per quanto riguarda Storia e Geografia si ipotizza l'utilizzo di un Data-base relazionale e un

Studenti al lavoro nel nuovo laboratorio di Pregnana Milanese

CAD che permettano la costruzione di una base di dati da aggiornare e consultare; di elaborare dati e costruire indici statistici per l'esame in serie di dati economici e demografici nel tempo e per la realizzazione di grafici e carte tematiche.
Amiga 2000 Education permette, inoltre, attività sulle lingue straniere (con l'uso di un programma di vocabolario inglese) al fine di poter realizzare esercizi di composizione in lingua con autocorrezione e autovalutazione e la costruzione di un vocabolario di termini conosciuti.
Per quanto riguarda la Matematica, è possibile l'impiego del computer nella rappresentazione di funzioni e studio della geometria analitica, nella soluzione di problemi con la formulazione di algoritmi tipo diagrammi di flusso per un programma di foglio elettronico.
Nel campo dell'educazione tecnica l'utilizzo di un CAD bi- e tridimensionale permetterà la creazione di disegni geometrici (proiezioni ortogonali, assonometrie e prospettive), la formazione di un archivio dati e la rappresentazione di dati statistici con possibilità di attività di progettazione, indagine e ricerca.
Infine, per l'Educazione artistica, si pensa di utilizzare un programma di grafica pittorica, di un CAD integrato ed eventualmente di un digitalizzatore di immagini.
Questo favorirà lo studio del disegno libero e la realizzazione di asimmetrie centrali e assiali nonché le prospettive; si potrà inoltre promuovere l'esercitazione di composizione grafica anche attraverso la manipolazione di immagini video fino a raggiungere tecniche di animazione di immagini. ▲

ON DISK è una rubrica mensile di quattro pagine che possono anche essere staccate e conservate, in queste pagine sono descritte tutte le informazioni dei programmi inclusi nel disco, complete di istruzioni, trucchi ecc... In questo spazio troveranno posto giochi, utility e tutto ciò che può fare Amiga.

• Utility

A-Mouse

A-Mouse è un'utility che permette di gestire i parametri del mouse. Oltre al controllo della velocità e della accelerazione, A-Mouse permette all'utente di gestire la velocità di risposta del mouse: nel Workbench, ad esempio, spostando il mouse di mezzo centimetro, il puntatore si sposta di un pixel; con A-Mouse è possibile ottenere un movimento di 640 pixel con lo stesso movimento. E' quindi possibile modificare i parametri per le proprie esigenze, a seconda che si debba fare un lavoro di precisione o un lavoro in cui serve il mouse veloce. La modifica di tutti questi parametri è piuttosto semplice, grazie al pannello di controllo che è stato progettato per poter essere richiamato da qualsiasi programma essendo abilitato a comparire su qualsiasi finestra o schermo; la modifica dei parametri del mouse, potrà avvenire dal proprio programma preferito, usando combinazioni di tasti per attivare il pannello di controllo o per disattivarlo. A-Mouse salva in un file le modifiche apportate che verranno automaticamente caricate la volta successiva. Il programma può essere lanciato sia da CLI che da Workbench, e una volta in esecuzione, tenta di caricare dalla directory DEVS: il file di configurazione "AMouse-configuration"; se non viene trovato, viene fatto lampeggiare lo schermo e attivato il pannello di controllo impostato con i valori di default. Sul pannello di controllo sono presenti tre slider per il controllo del puntatore: Speed, Pause a Pointer Prefs. Speed rappresenta il moltiplicatore usato per accelerare il mouse. L'intervallo consentito è tra 0 e 20. Il valore 1 è neutrale, mentre valori più alti consentono di accelerare il mouse. Impostando Speed a 0 il mouse viene rallentato. Pause è il ritardo d'accelerazione. Tecnicamente, è il numero di pixel per unità di tempo prima che il mouse acceleri. Anche questo parametro ha un intervallo tra 0 e 20. Con il valore zero, il mouse è sempre accelerato secondo il valore di Speed. Con valori superiori di Pause, bisogna dare un colpo più veloce al mouse affinché l'accelerazione abbia effetto. Pointer Prefs è il valore della velocità del puntatore impostato con preference e mantenuta come variabile di sistema. Può assumere i valori tra 1 e 4 e, quando viene lanciato, A-Mouse Pointer Prefs assume il valore di sistema. Ci sono, inoltre, cinque gadget in basso al pannello di controllo: Save, Reset, OK, AutoPop e Terminate A-Mouse. Save permette di salvare le modifiche apportate nel file di configurazione "AMouse-configuration" nella directory DEVS:, mentre Reset reimposta i parametri iniziali, quelli, cioè, salvati nel file di configurazione. Cliccando sul gadget OK, vengono accettati i parametri impostati e resi operativi. Il pannello di controllo viene chiuso. AutoPop serve per abilitare o disabilitare il pannello di controllo quando viene lanciato A-Mouse. Può quindi essere attivo (ON) o disattivo (OFF). Quando è ON, il pannello di controllo viene attivato appena lanciato il programma, mentre quando è OFF, il pannello non appare finché non viene attivato con l'apposita combinazione di tasti. Cliccando su Terminate A-Mouse il programma termina, viene cioè rimosso dalla memoria e vengono reimpostati i parametri di sistema. Vi sono tre combinazioni di tasti che controllano A-Mouse: Amiga-destro+Help che fa apparire il pannello di controllo, Amiga-destro+F9 che chiude il pannello di controllo ma non rimuove il programma dalla memoria (ha lo stesso effetto di OK) e, infine, Amiga-destro+F10 che imposta i parametri con i valori salvati nel file di configurazione (come se premessimo su Reset). Ancora una nota sull'uso di A-Mouse: visto che A-Mouse può essere richiamato da qualsiasi programma tramite Amiga-destro+Help, e quindi apparire su di uno schermo diverso da quello del Workbench, bisogna assolutamente assicurarsi di chiudere la finestra del pannello di controllo (tramite Amiga-destro+F9 o con OK) prima di chiudere lo scher-

mo sul quale si trova A-Mouse (la visita del Guru *non è gradita, vero?*). Di seguito, trovate alcuni valori indicativi per utilizzare A-Mouse.

	Speed	Pause	Pointer Prefs
Uso Generico	2	0	1
Uso per lavoro	8	1	2
Uso per lavoro di precisione	16	3	4

Boot Doctor

Boot Doctor è un'utility per la gestione dei boot block. Prima di iniziare a parlare delle varie opzioni e delle potenzialità del programma facciamo, un passo indietro per ricordare alcune sui boot block. Un boot block consiste, sostanzialmente, in un codice che viene eseguito quando si fa partire Amiga dal disco. Il boot block, che è situato nei primi due blocchi del disco, è, solitamente, standard per tutti i dischi, tranne che in due circostanze: quando un boot block è autocaricante, ossia carica automaticamente il programma senza cedere il controllo alla startup-sequence (è il caso di alcuni giochi protetti da copia), oppure quando il boot block è affetto da un virus (o quando è presente un antivirus). Il funzionamento di alcuni antivirus è quello di controllare se il boot block è standard o meno e, nel caso non lo sia, riscrivono il boot block con un codice standard o con un codice appropriato (si tratta di programmi che, una volta eseguiti in fase di boot, rimangono residenti in memoria e controllano successivamente, che, ogni disco inserito, abbia il boot block standard. Per ottenere un disco con boot block standard, si utilizza il comando CLI "Install" che riscrivere i primi due blocchi con il codice standard. Nel caso di giochi protetti, la riscrittura del boot block porta a conseguenze indesiderate: si rischia, infatti, di rendere inutilizzabile il gioco in quanto il boot block standard non andrà sicuramente a caricare il gioco presente sul disco. Bisogna stare quindi molto attenti a maneggiare e gestire i boot block. Boot Doctor non è una alternativa ad alcun antivirus, ma solo un programma per poter meglio maneggiare i boot block.

Il programma è molto semplice da utilizzare: con un doppio click lo facciamo partire e subito appare lo schermo con le otto opzioni disponibili. Una volta scelta l'opzione desiderata, appare un messaggio di conferma nella parte bassa dello schermo. Ad esempio, Press RMB to continue significa che bisogna premere il tasto destro del mouse per continuare (RMB è l'acronimo di Right Mouse Button). La prima opzione, Install Standard DOS Boot, riscrive il boot block con uno standard, qualunque sia il suo contenuto. Corrisponde esattamente al comando Install del CLI, per cui usatelo con cautela. La seconda opzione è Install [No DF1:]. Vi sono giochi che non funzionano correttamente quando è presente il secondo disk drive. Questo inconveniente è causato da AmigaDOS che alloca una certa quantità di memoria per ogni drive presente nel sistema privandoci, quindi, di preziosissima memoria; ciò che risulta fatale, soprattutto per chi possiede solo 512 Kbyte di memoria. Scegliendo questa opzione, viene installato nel boot block un apposito programmino che fa vedere al sistema solo il drive df0:. Prima di utilizzare questa opzione, si consiglia vivamente di fare una copia del vecchio boot block, tramite la sesta opzione di Boot Doctor. La terza opzione, Install [No FASTMEM] permette di disconnettere tutte le espansioni di memoria presenti nel computer. Questa opzione risulta particolarmente utile in presenza di programmi che, con l'espansione di memoria presente, non funzionano correttamente. Si tratta, per lo più, di programmi di vecchia data scritti in modo errato. Succede, infatti, che i suddetti programmi richiedono al sistema il primo tipo di memoria disponibile che, magari, deve essere riempito con dati che serviranno poi per grafica o suono, ossia per le funzioni svolte dai coprocessori che possono accedere solo alla memoria CHIP. Gli inconvenienti presentati sono di vario tipo. Di solito si nota questo inconveniente soprattutto nella grafica: potrà, ad esempio, non comparire il puntatore o i gadget o, al loro posto, comparire degli strani geroglifici incomprensibili (vi sarà sicuramente capitato che, al posto del puntatore, compaiano delle incomprensibili strisce verticali).

In questo caso le soluzioni sono due: o utilizzare programmi che cambiano la struttura interna dei file, tipo FixHunk o simili, oppure installare nel boot block un programmino che renda invisibile al sistema l'espansione di memoria presente. Questo è proprio il nostro caso; come per la seconda opzione, anche in questo caso viene installato un piccolo programmino che "nasconde" tutta la memoria FAST.

La quarta opzione Install [No DF1: or FASTMEM] è una combinazione della seconda e della terza opzione. La quinta opzione Save Boot as Executable è stata sicuramente progettata per coloro che sono interessati a vedere come sono fatti i boot block personalizzati o, molto più interessante, i virus. Questo sarà sicuramente utile a chi vuole imparare a progettare boot block personalizzati con grafica o suono incorporati. Una volta scelta questa opzione appare, in basso allo schermo, un messaggio di Boot Doctor che richiede l'inserimento del disco sorgente dal quale prendere il boot block da salvare come eseguibile:

```
Insert Source disk in drive
Press RMB to continue
```

Dopo aver premuto il tasto destro del mouse, comparirà un file requester per salvare il file su disco.

La sesta opzione, Save Archival BackUp of Boot, dovrebbe essere la funzione più utilizzata di Boot Doctor. Infatti, essa è la nostra arma contro i virus. Per ogni gioco dovremo fare una copia del boot block e salvarla su di un disco. Ad esempio, potremmo salvare i nostri boot block con Arkanoid.boot, Denaris.boot o Populous.boot. Dopo aver selezionato questa opzione, ci verrà richie-

Che cosa sono MFM e GCR?

Quei termini misteriosi termini sui dischi, spiegati una volta per tutte

Betty Clay

I disk drive sono cose affascinanti. Quando funzionano adeguatamente, gestiscono dati e software così bene che non c'è alcun motivo di chiedersi come lo fanno. Quando qualcosa va storto, tuttavia, la perdita è (o può essere) così grande che il funzionamento interno del drive diventa della massima importanza. Quando cominciai a chiedere come funzionano i disk drive, mi si rispondeva che i drive dovevano essere concepiti come scatole nere e che non si pretendeva che io li capissi, solamente che li usassi con gratitudine (qualcuno era ignorante come me, oppure stava diventando un po' maschilista?).

Finalmente un disk editor entrò nella mia vita e imparai a leggere e a interpretare la rappresentazione decimale di ciò che c'era sui miei dischi. Studiare attentamente i manuali dei dischi fornì qualche informazione e imparai a leggere le tracce delle directory, a trovare dove erano effettivamente collocate le varie cose sui miei dischi e qualche volta a salvare cose che altrimenti sarebbero andate perdute.

Fu attraverso gli scritti e gli insegnamenti di Mike Todd in Inghilterra e Dick Immers e Gerry Neufeld in Nord America che imparai a conoscere gli schemi di codifica dei dischi, gli header, gli header gap e tutte le altre sottigliezze che costituiscono la configurazione reale del disco. Non è che io avessi mai visto effettivamente un blocco di disco codificato in GCR, ma sapevo che cosa vi era scritto, come era stato scritto e come fosse riletto. Trovavo affascinante sapere cosa esattamente stava facendo il mio computer quando scriveva o leggeva il disco.

Uno dei miei primissimi progetti fu decifrare la configurazione dei dischi di questa nuova macchina. Fu un progetto meritevole, perché i dischi corrotti erano molto comuni a quel tempo e non avevamo DISKSALV per recuperarli. Fra le pochissime utility disponibili c'era un editor chiamato DISKED. Forse sono la sola persona cui piace veramente DISKED, ma esso mi ha consentito di imparare molto sulla struttura dei dischi in quei primi tempi e ci sono aspetti per i quali appare superiore a qualsiasi degli editor più popolari, apparsi in seguito e ancora oggi disponibili. Con DISKED e le scarse informazione del ROM Kernel Manual, ho potuto comprendere le informazioni degli header di ciascun blocco, le tavole hash usate nei blocchi di directory e ho potuto seguire un file dalla radice fino alla sua effettiva collocazione sul disco.
Ma per me, ciò non era ancora abbastanza. Stavo leggendo una rappresentazione di ciò che si trovava sul disco, ma ancora non sapevo cosa c'era realmente su di esso.
Il ROM Kernel Manual ci diceva come prima cosa che i drive di Amiga erano in grado di scrivere in formato GCR o MFM. Non dava ulteriori spiegazioni su di essi. GCR era familiare perché era usato negli altri drive Commodore. MFM, sebbene fosse un metodo antico per scrivere sui dischi, era nuovo per me. Andai cercando informazioni su questo metodo, ma trovai poco. Ovunque mi rivolgevo, si diceva "questo è un drive MFM" o "questo è un drive GCR" o "questo è un drive y", ma nessuno poteva mai spiegare esattamente che cosa significassero questi nomi.

Giorni di ricerca in libreria sono stati infruttuosi. Sembra esistano persone che conoscono e usano questi termini liberamente, ma ho potuto trovare pochissimi riferimenti sulla stampa. Gli sforzi per trovare le informazioni sulla stampa si sono dimostrati inutili. Le informazioni che ho scoperto erano a spizzichi e bocconi. Messe insieme, queste ghiotte notiziole sembrano essere sensate e corrette. Ecco cosa ho imparato.

Perché codificare?

Nel computer le informazioni sono conservate sotto forma di zero e uno, ma il computer le decodifica così che possiamo vedere caratteri alfanumerici.
Allo stesso modo, i dati sul disco vengono scritti come una stringa di zero e uno, o più precisamente come una sequenza di bit magnetizzati di ossido.

La testina di lettura/scrittura di un disk drive è una bobina. Quando la corrente passa attraverso questa bobina, crea un campo magnetico che viene registrato sul disco magnetico. Se la corrente va in una direzione, il bit è registrato con un "polo nord", viceversa con un "polo sud". Se non vi è registrato alcunché, il disco conterrà dei dati nulli in quel posto. I dati nulli si presentano solo negli intervalli tra i dati che Amiga pone alla fine di tutti i settori di una traccia.

Alcuni tipi di drive pongono dei dati nulli tra i settori, permettendo la lettura di un singolo settore da una traccia, piuttosto che di intere tracce come fa Amiga.

Quando la testina di lettura passa sopra una traccia, ignora i dati nulli. Se passa sopra un campo magnetico, si genera un impulso che viene interpretato come un dato. Generalmente un impulso viene considerato un on o bit di uno e la mancanza di impulsi è interpretata come un bit di off o zero. Non esistono dati nulli tra dati di bit. Essi si trovano solo tra settori. Il disco gira nel drive a una velocità costante e attentamente controllata di 300 giri al minuto. I drive di Amiga sono costruiti per mantenersi sempre entro variazioni del 1.5% di questa velocità. Ciò permette la lettura di una intera traccia di dati in circa 0.2 secondi, che comprende 11 settori e i loro header. I bit devono essere scritti sul disco in modo tale che il drive possa accuratamente determinare se ciascun bit è uno zero o un uno. I vincoli fisici del drive impongono che vi siano almeno due microsecondi tra cambi di polarità o bit di uno.

Al fine di rendere la lettura più accurata, sono stati sviluppati schemi di codificazione che rendono impossibile che due bit di uno si trovino troppo vicini per essere letti. Tali schemi stabiliscono anche regole per la scrittura dei dati, rendendo possibile la scrittura di bit che non possano essere interpretati come dei dati. Questi schemi realizzano due scopi. Il drive deve essere in grado di stabilire dove iniziano i dati se deve interpretare i numeri binari correttamente, ed esso deve anche essere capace di determinare in maniera accurata se ogni dato bit è uno zero o un uno.

L'inizio dei dati è marcato sul disco con un sync mark (contrassegno di sincronizzazione): una sequenza di bit che non è possibile incontrare all'interno di dati normali. Sui disk drive a 8 bit della Commodore, il sync mark consisteva in dieci o più bit di uno in fila, in quanto una volta codificato, un dato normale non poteva averne più di otto consecutivi. Su Amiga, un bit di dati non può mai avere più di quattro zeri e un sync mark. I bit di dati sono codificati in modi che impediscono la presenza di troppi bit di on o bit di off consecutivi, in modo che il drive possa leggerli in maniera accurata. Sebbene esistano molti metodi per codificare i dati, i due qui esaminati sono quelli che si trovano su Amiga, GCR e MFM.

La codifica GCR

Ci sono molte forme di codificazione GCR, ma quella descritta qui è quella usata nei drive ad 8 bit della Commodore. Amiga è in grado di usare la GCR, ma nessuno sembra usarla. C'è un vantaggio apparente: con la codifica GCR si possono immagazzinare in un disco più dati che con l'MFM che usiamo normalmente.

GCR sta per Group Code Recording, in quanto i bit sono raggruppati prima di essere codificati. Per usare questo sistema, ciascun byte viene separato in due nybble di quattro bit. Ciascun nybble è poi codificato secondo una tavola GCR, con la quale ciascun nybble di quattro bit sarà codificato come cinque bit GCR. Usando il metodo GCR non ci saranno mai più di otto bit di uno consecutivi o due bit di zero consecutivi.

Per n membri distinti di un insieme, ci possono essere 2 elevato n possibili disposizioni di essi. Così per un gruppo di quattro bit ci possono essere esattamente sedici modi in cui essi possono essere codificati. La tavola GCR, perciò, contiene sedici possibili traduzioni:

Decimale	Hex	Binario	Equivalente GCR
0	$0	0000	01010
1	$1	0001	01011
2	$2	0010	10010
3	$3	0011	10011
4	$4	0100	01110
5	$5	0101	01111
6	$6	0110	10110
7	$7	0111	10111
8	$8	1000	01001
9	$9	1001	11001
10	$A	1010	11010
11	$B	1011	11011
12	$C	1100	01101
13	$D	1101	11101
14	$E	1110	11110
15	$F	1111	10101

Quando si deve codificare un byte di dati, avviene un processo di questo genere:

	1	2
Prendi un byte di dati, per esempio 18 esadecimale		
Scrivilo in notazione binaria:	0001	0010
Converti il nybble alto, 0001, in GCR:	01011	
Converti il nybble basso, 0010, in GCR:		10010
Il byte risultante dalla concatenazione è:	0101110010	

Si noti che non compaiono mai più di due zeri in una riga. Nessuna voce della tavola consiste solo di uno e un esame di essa mostra che non ci possono essere più di otto uno consecutivi. Dal momento che i byte di 8 bit danno luogo a 10 bit GCR, in un computer a 8 bit ci saranno sempre dei bit in sospeso dopo il riempimento dello spazio concesso. Per questa ragione, un drive GCR tratta solitamente i byte a gruppi di quattro. Ciò produce 40 bit GCR che posssono essere trattati nella stessa modo di cinque byte regolari, e il numero è stato scelto perché 40 è esattamente divisibile in byte tradotti di 8 bit, essendo il minimo comune multiplo di 8 e 5.

Siccome 4 bit binari sono codificati come cinque bit GCR, l'efficenza è dell'80%. Cioè c'è l'80% dei dati che avrebbero

potuto esserci senza codifica, sia che si tratti dei vostri dati, che dell'informazione aggiuntiva richiesta dal drive. Ciò significherebbe consentire a un drive di contenere molte più informazioni di quante potrebbero essere immagazzinate con la maggior parte degli altri metodi. Tuttavia, siccome il GCR permette l'uso di 8 bit di uno consecutivi, il drive non può interpretarli a piena velocità. E' necessario scrivere e leggere a metà della velocità normale, per assicurare l'accuratezza necessaria. Quando la velocità di scrittura si abbassa fino a 4 microsecondi per bit invece dei 2 normali, la densità dei dati diviene solo la metà del massimo, consumando drasticamente i vantaggi dell'immagazzinamento.

Sui più vecchi drive sia delle macchine Apple che della Commodore, gli incrementi di capacità erano molto più importanti della velocità, così il GCR era usato da entrambi. Il GCR di Amiga è effettivamente compatibile con la versione Apple. Le tavole di ricerca sono le stesse, ma il funzionamento effettivo è leggermente differente.
Una volta chiesi a Jay Miner perché la tavola e la funzione GCR vennero inserite in Amiga senza essere mai utilizzate. Mi disse che le vollero per ragioni di compatibilità, ma che preferirono la più semplice e veloce codifica MFM per l'uso normale. Se doveste mai aver bisogno di esse, sono lì a disposizione e non occupano molta memoria.

La codifica MFM

Questo è il metodo di codifica usato correntemente sui floppy disk di Amiga e sulla maggior parte degli hard disk. Questo formato pone sia dati che bit di clock sul disco. Siccome c'è un bit di clock per ogni bit di dati, solo il 50% dei bit del disco rappresentano vere e proprie informazioni. I bit di clock vengono aggiunti quando i dati sono scritti, e cancellati man mano che vengono letti.

Con l'MFM, il sistema non richiede una tavola di ricerca perché le regole di codifica sono molto semplici. Se uno dei due bit di dati adiacenti è un bit di 1, tra i due viene inserito un bit di 0. Se due bit di dati adiacenti sono entrambi zero, allora un bit di 1 viene inserito tra di essi. Con questo sistema è certo che non ci saranno mai due bit di 1 adiacenti. Il sistema può quindi leggere e scrivere a due microsecondi per bit invece di quattro come in GCR.
Prendiamo ancora il numero $18 di prima e scriviamolo in notazione binaria:

	1 0 0 1 0
Poniamo i bit di clock	0 0 1 0 0
Così, codificato, risulta	0 1 0 0 1 0 0 1 0 0

La sincronizzazione determina anche l'interpretazione di questi bit a impulsi. Il computer presume che gli si passi un nuovo bit ogni due microsecondi, così ogni due microsecondi interpreta i dati come zero se, in quel lasso di tempo, non determina un cambio di magnetismo.
Quando si scrivono i dati sul disco, viene scritta un'intera traccia alla volta, ma viene trattata come se fosse divisa in settori. Ogni settore ha un header di sedici byte, ma questo header non viene normalmente visto dagli utilizzatori normali dei drive. Secondo il ROM Kernel Manual, gli header dei settori di Amiga, prima della codifica, contengono:

2 byte $00	rappresentati in MFM come $AAAA
2 byte $A1	un byte standard MFM di sincronia, che è $A1 codificato in MFM, ma senza un bit di clock
1 byte	che identifica il formato (per l'1.0 era $FF)
1 byte	che contiene il numero di traccia
1 byte	che contiene il numero di settore
1 byte	che contiene il numero di settori che mancano al termine delle operazioni di scrittura (si veda oltre)
16 byte	di informazioni per le riparazioni del sistema operativo
4 byte	di checksum dell'header
4 byte	di checksum dei dati

e questo è seguito dai 512 byte del settore che siamo in grado di leggere con un disk editor.
La codifica di questi byte è fatta in un modo curioso. I primi quattro byte dell'header sono codificati un byte alla volta. Prima si separano i bit dispari dai bit pari, poi sono fatti slittare (shift) a destra di una posizione, codificati (seguendo le regole MFM) e scritti sul disco. Poi vengono codificati e scritti i bit pari.

Se usiamo il nostro numero $18 come se fosse in uno di quei byte, verrebbe così elaborato:

18 =	1 2	esadecimale
	0 0 0 1 0 0 1 0	binario
	0 0 0 1	estraiamo i bit dispari
	0 0 0 1	shift a destra
	1 1 1 0	i bit di clock dispari
	1 0 1 0 1 0 0 1	scritto sul disco
	0 1 0 0	estraiamo i bit pari
	0 0 0 1	i bit di clock pari
	0 0 0 1 0 0 1 0	scritto sul disco

Così la rappresentazione completa del numero $18 sul disco sarebbe 1010100100010010. Tuttavia, questo è solo un'immagine del processo, in quanto, in realtà, lo schema di codifica deve prendere in considerazione l'ultimo bit dei dati codificati in precedenza (come avviene nell'esempio con l'ultimo bit dispari e il primo pari).

Ciò avverrebbe per ciascuno dei quattro byte dell'header e del segnale di sincronia. I successivi quattro byte (formato, traccia, settore e numero di settori alla fine) sono codificati come un singolo blocco, seguendo lo stesso modello: separazione dei bit dispari e di quelli pari, codifica dei bit dispari e scrittura, codifica dei bit pari e scrittura.
I 16 byte di informazioni per la riparazione del sistema operativo sono codificati come un singolo blocco, e nello stesso modo. Poi i quattro byte di checksum dell'header

sono trattati come un singolo blocco, seguiti dai quattro byte di checksum dei dati come un'altra unità.

Infine, viene codificato l'intero settore dei dati, di 512 byte, come una singola unità, prima i bit dispari, poi i bit pari. Per tutto tranne il singolo byte, la codifica e la decodifica vengono svolte dal blitter. I dati sono letti dal disco, decodificati dal blitter e posti nel buffer utente. E' probabilmente possibile evitare il passaggio attraverso il blitter e leggere le informazioni ancora codificate, ma non conosco nessuno che l'abbia fatto.

Blocchi che mancano al termine dell'operazione di scrittura

Forse questa non è realmente una parte del processo di codifica, ma i "blocchi che mancano al termine dell'operazione di scrittura" costituiscono una caratteristica curiosa dell'AmigaDOS. Quando si scrive su un floppy di Amiga, viene fornita alle testine del drive un numero di traccia su cui possono scrivere ed esse si muovono immediatamente verso la traccia data e cominciano a scrivere. Non aspettano un segnale di sincronia, né un settore particolare e non è probabile che scrivano sulla stessa parte della traccia usata in precedenza.

Questa è la ragione per cui molti dischi sembrano essere buoni la prima volta che li si formatta, mentre vengono registrati come "bad disk" al momento in cui si effettuano successive operazioni di scrittura. I primi accessi avevano usato parti buone della traccia; le parti corrotte si situavano nelle zone nulle poste tra i settori, così il disco veniva registrato come buono. In successive operazioni di scrittura, le informazioni cominciano in posti differenti della traccia, così la porzione corrotta non si trova più nell'intervallo nullo. Si ricordi che i settori di Amiga sono logici, non fisici. Non ci sono intervalli nulli tra i settori, mentre esiste un intervallo nullo piuttosto lungo alla fine dell'ultimo settore della traccia.
Le testine iniziano a scrivere nel punto esatto in cui hanno trovato la traccia. Se si tratta della prima volta che una traccia viene scritta, i settori verranno scritti nell'ordine esatto di successione. Le testine scriveranno un gruppo di byte nulli e poi scriveranno i settori.

Nell'header dei settori, il byte dei "settori al termine della scrittura" (l'offset) sarebbe così:

Settore: 0 1 2 3 4 5 6 7 8 9 10
Offset : 11 10 9 8 7 6 5 4 3 2 1

Questo numero di offset dice al drive quanti settori ci saranno prima dell'intervallo nullo. Quando questa traccia viene letta, le testine cominceranno a leggere non appena trovano la traccia giusta, senza ricercare alcun tipo di segnale. Una volta che questi settori siano stati letti, decodificati e posti nel buffer utente, questi offset li accompagnano. Supponete che quando la testina abbia iniziato a leggere si trovasse sopra il settore 5. Allora i dati letti apparirebbero così:

Settore: 5 6 7 8 9 10 11 dati nulli 1 2 3 4 5
Offset : 7 6 5 4 3 2 1 11 10 9 8 7

Si noti che il settore 5 deve essere letto due volte, perché è improbabile che la prima lettura cominci all'inizio del settore. In pratica, la prima lettura di questo settore verrà considerata, proprio per tale motivo, non valida.
Le informazioni verrano poi spostate nel buffer utente, ma non nel modo che ci si potrebbe aspettare. Il settore 6 verrà spostato all'inizio del buffer, seguito dagli altri fino al settore 11. I dati nulli vengono eliminati e i settori 1-5 faranno seguito al settore 11.

Quando queste informazioni verranno scritte sul disco, gli offset dei settori saranno modificati per riflettere il nuovo ordinamento dei settori:

Settore: 6 7 8 9 10 11 1 2 3 4 5 dati nulli
Offset : 11 10 9 8 7 6 5 4 3 2 1

Si noti che non compare mai uno zero come byte di offset perché questo offset deve essere letto prima del settore, ed esiste sempre almeno un settore da leggere prima dei dati nulli, nel momento in cui si incontra l'offset. Questo è molto importante perché la traccia potrebbe terminare con qualsiasi settore e il drive deve sapere quando interrompere la lettura. I sedici byte di informazioni per il sistema operativo normalmente non vengono usati. L'utente del disco non vedrà mai delle informazioni qui, ma è possibile che il programmatore vi inserisca dei dati, a condizione che tale spazio non sia richiesto da future revisioni del sistema operativo. Non è utilizzabile per protezioni contro la copia, ma potrebbe essere usato per immagazzinare informazioni di copyright, numeri di serie e così via. Qui si richiede la vostra immaginazione!
I drive di Amiga contengono ancora qualche mistero. Forse saranno tutti rivelati col tempo, forse no; e, naturalmente, l'intero capitolo degli hard disk deve ancora essere esplorato. Queste scatole nere rivelano i loro segreti lentamente, ma con l'ostinazione essi possono essere scoperti.

Ulteriori letture

Betty Clay, More on Amiga File Structure, Transactor, vol. 7, n. 6, May 1987, pp.72-73
Gelfland Abraham Grote, Amiga Disk Drive Inside and Out (first edition), Abacus,1988
Richard Immers e Gerald Neufeld, Inside Commodore DOS, Datamost,1984
Peck, Sanserrath e Deyl, Amiga ROM Kernel Manual, 2 voll., Commodore-Amiga, Inc.,1985
Thom Sterling, Fun with the Amiga Disk Controller, Amazing Computing, vol. I, n. 6, July 1986, pp.73-74

Scripit

La cosa migliore dopo il pane affettato... è ARexx

Steve Ahlstrom

ARexx è probabilmente il software più significativo comparso fino ad oggi per Amiga. Esso permette ai programmi dotati di una porta ARexx di comunicare e di controllare altri programmi con tale porta. Con esso è possibile integrare programmi provenienti da diversi produttori in un ambiente personalizzato che risponda ai vostri gusti.

L'intera comunità Amiga, ha visto il potenziale e il potere di ARexx. Persino la Commodore quando l'ha visto l'ha riconosciuto come una cosa buona (no, una grande cosa!) e, come risultato, ARexx farà parte del nuovo OS (2.0 N.d.T.), sebbene attualmente sia un prodotto commerciale.
Ora da Khalid Aldoseri proviene un altro prodotto che va mano nella mano con ARexx: Scripit. Non è un clone di ARexx, ma ha lo stesso impressionante potenziale, ma... si continui a leggere. Come il suo nome suggerisce, Scripit è un linguaggio di comandi per Amiga. Esso può fare praticamente ogni cosa che normalmente fate con il mouse e la tastiera, attraverso un file di comandi (script). E' un "processore di script" che legge uno script file ed esegue tutti i comandi del file in maniera sequenziale. Con ciò, può controllare qualsiasi programma attivando menu e gadget, come anche selezionare qualsiasi punto del programma, senza la necessità di una porta speciale.

Scripit può controllare programmi che ARexx non controlla. In sè non è un linguaggio di programmazione come ARexx. Tuttavia, possiede una porta ARexx, fornendo così ad ARexx il controllo su quei programmi che non hanno porte ARexx. Scripit è un insieme di programmi: un Recorder, un esecutore di script, un Lister, un programma di interfaccia con il Workbench e contiene altri vari aiuti alla programmazione.
L'interfaccia con il Workbench permette a Scripit di impradronirsene e di controllarlo pienamente. Potete selezionare le icone, aprire e chiudere i drawer (cassetti), muovere le icone, lanciare programmi, qualsiasi cosa possiate fare con il mouse (e la tastiera), ma da uno script.
Se siete preoccupati di dover indicare le icone mediante delle coordinate, dimenticatelo; potete riferirvi alle icone mediante il nome. Scripit ha un set di comandi molto ricco. Ha estese possibilità di manipolazione delle finestre e degli schermi, che vi permettono di muovere, ridimensionare, chiudere e mandare sullo sfondo o in primo piano schermi e finestre. Viene fornito un insieme completo di comandi per la console che vi consentono di aprirle e chiuderle e inviare ad esse qualsiasi comando ANSI.
Ha anche un insieme di comandi grafici che permettono il disegno su qualsiasi schermo di qualsiasi programma o l'inserimento di testo in una finestra.

Quando lo si seleziona dal Workbench, appare un file requester. Esso cercherà inizialmente la directory SCRIPIT:, e per default la directory corrente. Di là, basta selezionare lo script che volete eseguire. Potete anche creare un'icona per gli script (qualsiasi icona "project" andrà bene) e mandarli in esecuzione con una doppia selezione.
Quando viene lanciato dal CLI, Scripit consente molte opzioni sulla linea di comando:

```
Scripit [flag] [scriptfile]
```

Flag:
-v seleziona il modo Verbose (prolisso)
-d seleziona il modo Debug
-x carica Scripit in modo ARexx:
Scripit -x [nomeporta]
dove "nomeporta" è il nome della porta ARexx di Scripit
-c Compila lo script
-w disabilita gli avvertimenti (Warning)

Scripit può compilare gli script, riducendo del 30-50% lo spazio del disco necessario per lo script; gli script compilati vengono lanciati con XIT (incluso nella release di Scripit in distribuzione). E' molto più piccolo di Scripit e, pertanto, viene caricato più velocemente. Scripit compilerà automaticamente uno script non compilato prima di eseguirlo. Se, per qualche ragione, non si può leggere l'intero script tutto in una volta, compilerà ed eseguirà lo script una linea alla volta. C'è anche un modo "live". Digitando SCRIPIT * dal CLI, potete inserire i comandi direttamente.

La raccolta di Scripit

Il Recorder è un registratore di eventi Intuition. Registra esattamente che cosa fa l'utente e lo salva in un file che è esso stesso uno script. Quando verrà eseguito da Scripit, ripeterà tutte le azioni che sono state eseguite quando è stato creato. Il Recorder non è solo un registratore del mouse e della tastiera; esso registra tutti i messaggi che

vanno da Intuition alla finestra attualmente attiva. Se cambiate la finestra attiva mentre registrate, il Recorder aggiungerà nello script i comandi necessari per cambiare la finestra attiva.

Mentre il Recorder è acceso il LED dell'alimentazione lampeggerà per ricordarvi che siete nel modo registrazione. Dopo aver spento il Recorder, potete editare il file risultante proprio come qualsiasi altro file di testo. Il Recorder permette molte opzioni. Potete aggiungere automaticamente commenti allo script file; fargli registrare i numeri dei gadget e dei menu, invece dei nomi; evitare di far lampeggiare il LED quando registra; disabilitare la registrazione degli intervalli temporali, i MouseMove e gli Intuitick.

Il fomato del comando è il seguente:

```
recorder [-flag] [scriptfile]
```

Flag:
-c aggiunge Commenti allo script
-w non apre la Window di visualizzazione
-l disabilita il segnale di attività del LED dell'alimentazione
-n usa Numeri per i menu e i gadget invece di stringhe
-t disabilita la registrazione degli intervalli Temporali
-m disabilita la registrazione dei MouseMove
-i disabilita la registrazione degli Intuitick

Il Lister è un programma per generare elenchi relativi a schermi, finestre, menu e gadget. Esso aiuta ad identificare i nomi o i numeri delle voci da controllare. Lanciatelo e otterrete un elenco degli schermi del sistema. Potete vedere le finestre di uno schermo specifico digitando le prime lettere del nome dello schermo come primo argomento per Lister. Potete proseguire specificando finestre, menu, voci o sottovoci di menu, gadget. Il suo formato è:

```
Lister [schermo] [finestra] [menu] [voce] [sottovoce]
```

oppure:

```
Lister [schermo] [finestra] -g [gadget]
```

XitLoadWB è il programma di interfaccia con il Workbench ed è ciò che vi fornisce pieno controllo sul Workbench. Per usarlo, premettete a "LoadWB" nella vostra startup-sequence "XitLoadWB". Al prossimo boot XitLoadWB prenderà il controllo del Workbench.
Esiste una varietà di comandi Workbench: Selects, Open, Close, Deselect, ShiftSelect, ShifOpen, SnapShot, CleanUp, ReDraw, Info, Menu, Drag, DragTo e DragOver.
Scripit possiede anche la caratteristica "AutoScript". Potete costruire uno script che verrà lanciato automaticamente. Invece di digitare (nel CLI) "Scripit nomescript", potete semplicemente digitare "nomescript". E' simile ad ARexx quando viene usato in congiunzione con WShell: RX non è necessario.

Alcuni esempi

Mi rendo conto di avervi detto troppe cose, ed ora potreste essere totalmente confusi! Qui sotto ci sono alcuni script di esempio commentati.
Questo è uno script che mostra come lanciare un programma e compiere un insieme di azioni:

```
runback "dh2:Access/Access!"
waitfor "Access"
menu "Phone" "Re-Dial"
```

Le virgolette non sono importanti, a meno che non inseriate degli spazi negli argomenti.
Se avete bisogno di specificare degli argomenti CLI quando lanciate il programma, usate:

```
runback "nomedelprogramma argomento..."
```

Quello che segue è una versione più ampia dell'esempio precedente:

```
runback "dh2:Access/Access!_Supra"
waitfor Access #5
menu Phone Re-Dial
runback "dh2:Access/ShowModem -sAccess"
waitfor Access "Supra Modem"
window moveto 640,0
```

Quello che segue lancia Wordperfect, ridimensiona la sua finestra perché occupi metà dello schermo, ordina a WP di aprire una nuova finestra, la ridimensiona perché diventi la metà di quanto era in precedenza e la sposta sul fondo dello schermo, lasciando WP pronto per l'uso con due finestre.

```
runback "WP:WP"
waitfor Workbench WordPerfect
window resizeto 640,100
wait
menu Project New
waitfor Workbench "WordPerfect 4.1 - Doc 2"
window resizeto 640,100
wait
window moveto 0,100
```

Come spero conveniate, Scripit è uno strumento molto potente per rendere il nostro computer più facile da usare, automatizzando i compiti più ripetitivi. Lo diventa anche più quando viene guidato da ARexx per controllare quei programmi che non lo supportano ancora.

Scripit è un programma di Khalid Aldoseri liberamente distribuibile, ma soggetto a copyright. E' disponibile sul disco che accompagna la rivista o su Compuserve, dove si possono trovare script d'esempio e aiuto da parte di Khalid e Brian Bartlett.

L'allocazione della memoria su Amiga

I problemi di frammentazione della memoria

Eric Giguère

Eric Giguère è un ricercatore dell'Università di Waterloo, dove progetta un compilatore C conforme ANSI.

L'allocazione delle risorse è una delle funzioni più importanti in un sistema operativo. Assume un'importanza ancora più grande in un sistema multitasking come Amiga, dal momento che il multitasking implica la condivisione della CPU fra più task o processi. Anche la memoria è una risorsa che deve essere condivisa dai task, sia quando i programmi vengono caricati che quando vengono eseguiti. Questo articolo riguarda l'allocazione dinamica della memoria; cioè le richieste di memoria che vengono fatte dai programmi mentre sono in esecuzione.

Un esempio

Molti programmi C richiedono blocchi di memoria al sistema di tanto in tanto. Il programma che accompagna questo articolo ne costituisce un esempio. Non è particolarmente interessante: tutto ciò che fa è costruire un'enorme lista per poi tagliarla a metà. Con ciò renderà probabilmente inutilizzabile il vostro Amiga. Digitate il suo nome e provatelo, ma, per favore, ricordate di salvare qualsiasi cosa su cui stiate lavorando prima di lanciarlo! Quando avrà finito dovrete fare un boot. Nonostante la sua semplicità, il programma tocca tre questioni collegate all'allocazione della memoria:

1- Allocazioni equilibrate della memoria: le vostre richieste di memoria sono troppo grandi o troppo piccole?

2- Restituzione della memoria: quando i vostri programmi terminano, restituiscono tutta la memoria al sistema? In quale ordine la memoria viene restituita?

3- Gestione delle condizioni di memoria scarsa: se il sistema non può soddisfare una richiesta di memoria, il vostro programma andrà in crash?

Per trattare queste questioni e stabilire perché il programma fa ciò che fa, dobbiamo dapprima occuparci di alcuni concetti di fondo.

La memoria e Amiga

L'allocazione di memoria su Amiga è controllata da Exec, il livello più basso del sistema operativo di Amiga. Il capitolo 6 dell'Amiga ROM Kernel Reference Manual: Exec descrive le varie routine di sistema disponibili al programmatore per l'allocazione di memoria. Le routine che sono usate più frequentemente dai programmatori sono AllocMem( ), per allocare un blocco di memoria e FreeMem( ), per restituire un blocco allocato in precedenza al sistema.
AllocMem( ) ha la forma:

```
void *AllocMem( long lunghezza, long flag );
```

dove lunghezza è la quantità di memoria desiderata (in byte) e flag indica le diverse opzioni di memoria (che si trovano nel file include exec/memory.h):

```
MEMF_CHIP     alloca dalla CHIP RAM
MEMF_FAST     alloca dalla FAST RAM
MEMF_CLEAR    azzera la memoria allocata
```

Ciascuna opzione corrisponde all'impostazione di un bit nel parametro flag, così si può fare un OR logico fra di essi, ad esempio:

```
MEMF_CHIP|MEMF_FAST
```

permetterà al sistema di allocare memoria sia dalla FAST che dalla CHIP RAM.
Viene restituito un puntatore all'inizio del blocco di memoria allocato, se AllocMem( ) non riuscisse a soddisfare la vostra richiesta di memoria restituirà NULL e il programma dovrebbe prendere adeguate misure per uscire da questa condizione di memoria scarsa.
Anche FreeMem( ) richiede due argomenti:

```
void Freemem(void *ptr, long lunghezza);
```

dove ptr punta all'inizio del blocco che è stato allocato in precedenza con AllocMem( ) e lunghezza è la lunghezza del blocco.
Una tipica sequenza di allocazione e restituzione di memoria potrebbe apparire così:

```
char *ptr;
ptr = (char *)AllocMem(100L,MEMF_CHIP|MEMF_CLEAR);
/* fa qualcosa */
```

```
FreeMem( (void *) ptr, 100L);
```

Ci sono due punti importanti da ricordare:

1- Non tentate mai di restituire memoria che non sia stata allocata con AllocMem( ).

2- Non usate mai memoria che sia stata liberata con FreeMem( ).

Entrambe le cose possono causare un crash del sistema. Il secondo punto è estremamente importante perché è un errore comune riutilizzare la memoria che è stata già liberata. Se il sistema l'ha destinata a qualche altro processo, voi state calpestando i dati che appartengono a quel processo.

Dettagli tecnici

Exec deve tenere traccia di quale memoria sia disponibile prima di poterla distribuire. Fa questo mantenendo una lista circolare chiamata free list. Questa free list consiste di tutti i blocchi di memoria contigui attualmente non allocati (o non usati dal sistema).
Quando viene fatta una richiesta di memoria, Exec esamina la free list per trovare il primo blocco disponibile che sia abbastanza largo da soddisfare la richiesta. Se il blocco che trova è troppo largo, Exec lo dividerà e restituirà la porzione non usata alla free list. Quando un processo restituisce un blocco, viene semplicemente reinserito nella free list.
La lista è conservata a sua volta nella memoria libera. I primi quattro byte di un blocco libero contengono un puntatore al prossimo blocco libero; un secondo gruppo di quattro byte contiene la lunghezza di quel blocco.
Questo spiega perché AllocMem( ) alloca sempre la memoria in multipli di otto byte, e perché la lunghezza minima di un blocco è di otto byte. Se le vostre richieste di memoria sono troppo piccole, potreste devastare più memoria di quanta ne andiate allocando.

In ogni modo, questi dettagli non è detto che vengano rispettati nelle revisioni future: sono presentati qui solo per pura curiosità, non fate affidamento su di essi!
La frammentazione della memoria avviene quando molti piccoli blocchi di memoria libera sono separati da blocchi di memoria allocata.
Exec tenta di fondere i blocchi adiacenti di memoria libera, quando può. I blocchi ampi rendono più facile soddisfare le successive richieste di allocazione.
Questo è tutta l'economia che Exec può fare: è un compito impossibile spostare dei blocchi liberi per riunificarli, in quanto i blocchi di memoria allocata che si trovano fra di essi non possono essere spostati.

Le regole per l'allocazione della memoria

1- Tentate di tenere le richieste di allocazione della memoria al minimo.
2- Controllate sempre se le vostre richieste sono state soddisfatte. E' meglio terminare gentilmente (liberando tutte le risorse) che andare in crash per mancanza di memoria.

3- Liberate tutta la memoria che avete allocato prima di terminare il programma. Exec non tiene traccia della memoria che viene allocata: conosce solo quella che sia stata esplicitamente liberata. Terminare un programma non è sufficiente per restituire tutta la memoria allocata alla free list.

4- Se potete, usate la memoria FAST prima della memoria CHIP. La memoria CHIP è un lusso che è meglio risparmiare per la grafica e l'audio. Naturalmente, non tutti gli Amiga hanno la memoria FAST, così usate la combinazione MEMF_FAST | MEMF_CHIP: la memoria FAST sarà controllata per prima.

Il programma

Ora siete in grado di comprendere che cosa fa il programma che accompagna l'articolo. E' un esempio estremo di frammentazione della memoria.
Dapprima alloca tutta la memoria che può in una lista doppia. Poi procede a liberare tutti gli elementi pari della lista. Dopo averlo fatto, tenta di allocare un blocco di memoria leggermente più ampio di un singolo elemento della lista. Tuttavia non ci riesce a causa della frammentazione; tutti i blocchi liberi sono troppo piccoli.

Sul mio Amiga 2000, il programma mi ha dato come risultato che avevo 1 MByte di memoria FAST libera, sebbene fosse impossibile allocare un blocco di soli nove byte!
Non solo, ma il programma non libera tutta la sua memoria quando termina, lasciando Amiga in uno stato di frammentazione permanente. Ecco perché dovrete effettuare il boot, specialmente se lo lanciate usando la memoria CHIP. Senza dubbio, questo esempio è piuttosto innaturale; gli elementi della lista sono lunghi esattamente otto byte e il programma effettua l'allocazione della memoria in maniera lineare. Ma è una buonba dimostrazione di ciò che può accadere nei casi peggiori.

Note conclusive

L'allocazione della memoria non è particolarmente complessa se confrontata con altri aspetti del sistema operativo di Amiga, ma è cruciale per il funzionamento della macchina. Usate le routine di Exec propriamente e in maniera consistente, eviterete molti patemi d'animo.

Maggiori informazioni sugli schemi di allocazione della memoria si possono trovare in The Art Of Computer Programming, Volume 1: Fundamental Algorithms, di Donald Knuth, pubblicato dalla Addison Wesley, e nella maggior parte dei libri sui sistemi operativi.

Dal C all'assembler... e ritorno

Chiamare codice assembler dal C è facile

Peter Booth

Anche se il C è un linguaggio molto potente per la programmazione seria di Amiga, ci sono delle volte in cui conviene visitare per un momento l'assembler per garantirsi un incremento delle prestazioni.
Per essere in grado di scrivere "codice misto" dovete sapere come gestire i meccanismi di passaggio dal codice del programma C al codice assembler, la buona notizia è che, avendolo visto una volta, potrete capire che non è affatto difficile!

I dettagli necessari sono sempre forniti dal manuale del compilatore, ma, per essere onesto, le spiegazioni sono solitamente scritte in un modo comprensibile solo se si conosce almeno un poco che cosa, fondamentalmente, sta succedendo; di qui questo piccolo articolo, una spiegazione priva di fronzoli, di basso profilo, basata sul Lattice per spingervi a cominciare!

Ho scritto due moduli di codice esemplificativo: un breve programma in C (che appare su disco con il nome WordCount.c) che esegue una chiamata a una routine in linguaggio assembly e il codice assembler (WordCount.asm). Vi spiegherò che cosa fanno più tardi, ora dobbiamo parlare un poco delle "magie" del compilatore C.

Partendo dal C

Quando ponete una chiamata a una routine, chiamiamola AsseblerCode( ), in un programma sorgente in C, il compilatore usa l'istruzione del codice sorgente per generare un riferimento alla routine AsseblerCode( ).

Il compilatore aggiunge un carattere di sottolineatura all'inizio del nome della funzione, così la chiamata alla funzione C AsseblerCode( ) implica che il linker cerchi una routine chiamata _AsseblerCode( ), ed è questa la routine che dovete fornire!

Il codice esatto che i vari compilatori C producono quando incontrano una chiamata ad una funzione varia e, di nuovo, è sempre spiegato nel manuale del compilatore.

Tutto ciò che vi occorre sapere, tanto per cominciare,è che ogni parametro presente nella chiamata della funzione, alla fine, viene posto sullo stack e che in seguito viene effettuata la chiamata alla routine appropriata.
Non intendo discutere del passaggio dei parametri in questo articolo, perchè ciò si aggiunge proprio al numero di cose che potrebbero andare male, se state sperimentando per la prima volta; ma suppongo si debba ricordare che, con il Lattice, è anche possibile istruire il compilatore perchè passi alcuni degli argomenti mediante i registri e, quando ciò avviene, il Lattice etichetterà il nome del simbolo con un "@" invece che con un carattere di sottolineatura.

In conclusione, sta a voi consultare il manuale del vostro compilatore per vedere quali convenzioni è necessario adottare.

Il lato assembler

Superato il codice C, il prossimo passo è scrivere del sorgente in linguaggio assembly e assemblarlo per produrre codice oggetto, che si possa cioè passare al linker (e non sia del codice eseguibile direttamente). Avete bisogno di usare un paio di direttive assembler chiamate XDEF e XREF per far funzionare le cose senza problemi.

XDEF e' una direttiva assembler usata per rendere le etichette visibili ad altri moduli in fase di link: se la dimenticate il linker non sarà in grado di risolvere i riferimenti a _AsseblerCode( ) presenti nel modulo di codice C.
XREF percorre la via opposta: dice all'assemblatore che l'informazione necessaria sull'elemento in questione sarà importata quando il modulo in linguaggio assembly verrà sottoposto al linker.

La maggior parte degli assembler pongono un limite al numero di caratteri significativi e lo standard ANSI richiede che il compilatore provveda a solo sei caratteri nel caso di riferimenti esterni (sebbene la maggior parte ne gestira' un numero maggiore). O controllate prima, oppure xxnonxx usate nomi lunghi per le funzioni e le variabili cui faranno riferimento altri moduli.

Se avete impostato tutti i vostri riferimenti e le direttive in maniera corretta, il resto è facile. Compilate il vostro sorgente C, assemblate il vostro codice assembly e poi usate Blink per collegarli al codice di startup.

(segue a pag. 45)

RKM: Includes & Autodocs

Il nuovo ROM Kernel Manual: è all'altezza delle aspettative ?

Peter Booth

Titolo: Amiga ROM Kernel Reference Manual: Includes & Autodocs
Editore: Addison Wesley
Altri dati: 768 pp., in brossura, ISBN 0-201-18177-0, L.71000 (circa)

E' il primo dei nuovi manuali di riferimento tecnico ad essere pubblicato... ma prima di vedere che cosa offre, è bene parlare dell'ordinamento usato nella nuova edizione che aggiorna la documentazione alla versione 1.3 del software di sistema.
Come ci si aspettava, c'è ancora un "Hardware manual" separato, ma c'è anche un nuovo volume "Libraries and Devices" che contiene il materiale didattico su ROM Kernel, Exec, Intuition e Workbench.

RKM: il primo volume

Tutti i file include, gli autodoc delle funzioni e le voci diverse, come le note IFF, ora sono state spostate in un volume separato chiamato "Include & Autodocs". Destinato a divenire noto come "I&A", esso, come ci si potrebbe aspettare, richiede molte ore di lettura notturna e, senza dubbio, i soldi qui sono spesi bene.

La prima sezione comprende l'elenco delle librerie e aggiorna pienamente questa area della documentazione rispetto al nuovo software di sistema: solo l'accesso a questo materiale è in grado di gratificare la maggior parte dei fan di Amiga. L'elenco delle funzioni è organizzato alfabeticamente, libreria per libreria: è un ordinamento migliore del precedente, in quanto aiuta a raggiungere una valutazione complessiva delle funzioni disponibili in una data libreria. Al termine del manuale compare un esteso indice delle funzioni, così è ancora abbastanza facile trovare ciò che si cerca.

Dopo i dettagli sulle funzioni, viene la sezione dedicata ai device. Non c'è materiale didattico (che è tutto nel volume "Libraries & Devices"), è solo il semplice elenco delle richieste per i device e così via. Ci sono, tuttavia, alcune nuove gemme nascoste nelle note alle varie richieste per i device. Per esempio, quanti di voi sapevano che dopo avere effettuato un AbortIO( ) con il device seriale è in genere necessario fare un WaitIO( )?

La breve sezione con l'elenco relativo a dischi, cia, potgo e misc resource, è seguita dai particolari sugli imponenti file include del C e del linguaggio assembly e da una sezione che fornisce il codice sorgente per una libreria e un device d'esempio. Che cos'altro ricevete per i vostri soldi? Bene, sono fornite alcune maneggevoli tavole di riferimento che forniscono dettagli sulle librerie Amiga, gli offset delle loro funzioni, i prefissi dei file include per il linguaggio assembly, e ancora gli offset dei singoli elementi di ogni struttura. C'è una mappa dei registri hardware e una tavola dei riferimenti incrociati dei file include del C.

Il supporto IFF

Il materiale IFF presentato originariamente nel vecchio manuale Exec, si è accresciuto molto in ampiezza (sia come ampiezza fisica della stampa che come contenuto). Ora fornisce alcune utili note introduttive, il documento EA IFF 85 e i particolari sulle specifiche del Form originale. Sia SMUS che 8SVX, relativi al campionamento dei suoni, sono particolarmente ben trattati.

La discussione su FTXT include ancora i due semplici algoritmi scritti in stile Pascal per estrarre i testi da un file IFF. Se non li avete visti la prima volta, vale la pena di osservarli adesso: la prima versione è ricorsiva e mostra come le cose dovrebbero essere fatte, la seconda versione è un "metodo di lettura negligente": esso mostra, cioè, come le cose vengono fatte di solito!

Ora sono anche trattati Form e Chunk registrati da terzi, fra cui SMUS. PAN, ACBM, ANIM (Aegis/Sparta) e PGBT (Lattice) e c'è una buona scelta sia di codici esempio che di aiuti in stile didattico. Qualcuno mi ha fatto notare che è notevole l'assenza dei particolari IFF relativi a Music X della Microillusion. Non è un'omissione, è solo che MSCX è un Form IFF privato e non registrato e non vuole essere un formato di uso generale per i file dei sequencer (i particolari sui Chunk necessari sono tuttavia forniti nel manuale di Music X).

E' attendibile?

Ed ora la domanda più difficile: ci sono degli errori? Bene, ci sono certamente dei refusi tipografici, come parole mancanti in alcune nelle note sulle funzioni, ma nella

maggior parte dei casi il significato è abbastanza ovvio. Alla prima lettura nessuno può assorbire e confrontare nei particolari ogni funzione e ogni richiesta per i device, così non ho intenzione di assicurarvi in via definitiva che esso contiene più o meno errori dei precedenti manuali (sebbene sia sicuro che in generale le cose siano migliorate). Al supporto tecnico della Commodore non è stato ancora riportato alcun bug della documentazione delle funzioni, ma, per essere realisti, è destino che ne siano saltuariamente scoperti qua e là: è qualcosa con cui si deve convivere!

Più in generale, tenendo presente la complessità di Amiga e della sua documentazione, l'Addison Wesley e la Commodore dovrebbero riunire i loro sforzi per quanto riguarda le correzioni e le aggiunte. Gli aggiornamenti dei manuali saranno apparentemente presentati nella pubblicazione di supporto tecnico "AmigaMail", ma dal momento che questa è riservata agli sviluppatori registrati, ci saranno schiere di utenti del RKM che non potranno mai vedere gli aggiornamenti. Sono sicuro che dopo che i manuali saranno stati usati per un certo tempo, la maggior parte degli utenti non sarebbe contraria a spendere una piccola cifra per un volume (si spera sottile) di rettifiche, che riguardi gli errori e le omissioni dei vari manuali.

Con gli orrori della documentazione di transizione fra 1.1 e 1.2 ancora freschi nella mia mente, e sapendo che l'1.4 [2.0, N.d.T.] è proprio dietro l'angolo, ho ritenuto molto importante domandare alla Commodore come intenda gestire l'aggiornamento della documentazione dei RKM quando apparirà l'1.4. La risposta è stata che saranno probabilmente pubblicati uno o più supplementi contenenti le novità più importanti. Ciò per quanto riguarda il futuro del manuale "Includes & Autodoc". Che cosa ne penso? Non ho certo intenzione di stroncarlo; questo volume contiene una grande quantità di importanti informazioni.

Ci sono molti fra noi patiti di Amiga ai quali non sembra mai di ricevere dalla Commodore tutte le informazioni tecniche che desiderano. In teoria possiamo accedere agli autodoc su disco e alle note tecniche, ma in pratica non è così facile e un numero sorprendente di noi è sopravvissuto usando i manuali originali della Addison Wesley, con l'aggiunta dei saltuari comunicati e delle note di aggiornamento della Commodore (e l'aiuto di alcune persone introdotte negli ambienti giusti).

Solo l'avere una documentazione aggiornata sulle funzioni e sui device rende questo volume un investimento conveniente, e non c'è alcun dubbio che è un assoluto must per tutti gli utenti seri.

(segue da pag. 43)

Dal C all'assembler... e ritorno

Il programma di esempio WordCount

Ora alcune note sull'esempio vero e proprio. E' un programma basato sul CLI che conta le parole presenti in un file di testo ASCII. Il programma dà un rapido sguardo al file specificato sulla linea di comando, controlla la sua lunghezza e poi alloca un buffer di conseguenza.

Legge il file nel buffer e immediatamente passa il controllo alla routine in linguaggio assembly che analizza il file parola per parola. Quando la routine in linguaggio assembly passa il controllo al modulo C, viene stampato il numero di parole trovate e il programma termina ed esce.

Vedrete le mie istruzioni XDEF e XREF all'inizio del modulo in linguaggio assembly. Il Lattice C si attende che voi preserviate il contenuto dei registri D2-D7 e A2-A6, così questo è ciò che la mia routine fa come prima cosa. Poi cattura il contenuto di una coppia di variabili C (il puntatore al buffer e la lunghezza del file), inizializza il contatore delle parole e poi si "tuffa" in un loop doppio che localizza l'inizio e la fine delle parole.

Il codice in linguaggio assembly controlla i gruppi di caratteri nel campo a-z/A-Z e ritiene una parola qualsiasi insieme di questi caratteri delimitato da qualsiasi altro carattere. Ogni volta che trova una parola, incrementa una variabile chiamata _wordcount, che è la variabile C di tipo long, wordcount.

Dal momento che il programma è solo un esempio, non ho ottimizzato il codice assembler, sebbene anche così non sia malvagio quanto a velocità. Un file di testo di 100K, una volta caricato, viene analizzato in meno di un secondo; tentate di batterlo usando solo il C!

Il gioco del Caos

E' un frattale ? E' Caos ? O è...

Robert P. Hale e Victor A. Wagner Jr.

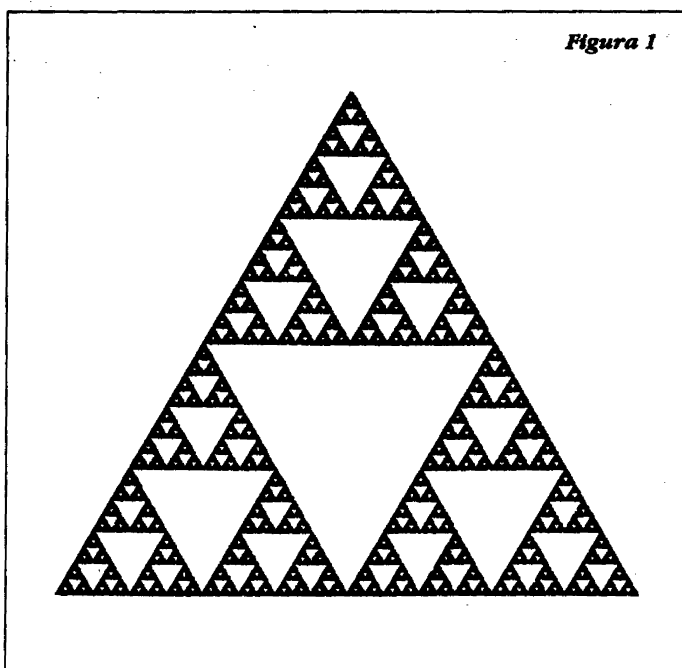
Liberamente definito, il Caos si pone tra il prevedibile e il casuale. I fenomeni caotici sono circoscritti, ma non possono essere simulati. La scoperta di algoritmi semplici che producono complessità infinita è uno dei motivi che stanno dietro lo studio del Caos.
L'introduzione di Nova: The Strange New Scienze of Chaos, un programma televisivo della WGBH a Boston, era qualcosa del genere:

"Stiamo per giocare al gioco del Caos.

1- Prendete 3 punti agli angoli di un grande triangolo e segnateli. Poi prendete un punto casuale in qualsiasi posizione della pagina.

2- Lanciate un dado, per scegliere uno degli angoli, e segnate il punto medio fra il punto corrente e l'angolo.

3- Ripetete il passo 2 molte volte.
Con ciò, vi aspettereste di riempire la pagina con una miriade di punti in maniera del tutto casuale. Invece, ciò che otterrete è questo..."

Figura 1

Ciò che apparve sullo schermo della TV, quasi per magia, è una trama simile a un pizzo costituita interamente da triangoli di varie ampiezze.

Un programma e molte domande

Dal momento che questa è una rivista su Amiga, potete probabilmente immaginare che cosa passò nella nostra mente quando vedemmo il programma. Le prestazioni di Amiga, come la grafica semplice da usare e la disponibilità di molti colori, ci aiutarono molto nello studio di questo problema. Lo sviluppo è andato avanti con uno stile del tipo "Che cosa avverrebbe se facessimo così?". Nel comunicarvi le informazioni, abbiamo tentato di mettere in ordine il tutto.

Scrivere un programma in C che insegni ad Amiga come giocare al gioco del Dr. Barnsley è piuttosto semplice:

```
for (i=0; i < NumeroArbitrariamenteGrande ; i++)
{
  r = rand() % 3; /* r = 0, 1 o 2 */
  x = (x + Angolo[r].x)/2;
  y = (y + Angolo[r].y)/2;
  WritePixel( rp, scala(x), scala(y));
}
```

Senza alcun dubbio, il triangolo sfaccettato appare sullo schermo.
Ma questo fa nascere più interrogativi di quanti ne risolva. Il primo, naturalmente, è perche? Non è così ovvio... non ancora, almeno. Altri interrogativi si presentano, comunque:

- Quali effetti ha la posizione del punto iniziale?
- In che modo colorare i punti per descrivere dove si stanno dirigendo?
- Che cos'è l'area parzialmente colorata?
- Che cosa accadrebbe con più lati?
- Che cosa accadrebbe con un rapporto diverso da 1/2?
- E' un frattale?

La posizione del punto di partenza

Non sembra importante: se il punto è fuori dal triangolo, si

muove rapidamente al suo interno. Queste rende la procedura un attrattore, nella terminologia del Caos.
Se il punto iniziale si trova in un'area vuota entro il triangolo, scompare rapidamente in una piena. In questo caso, con "rapidamente" si intende meno di dieci turni circa. Ci deve essere qualche ragione che spinge i punti a stare fuori dalle aree bianche.

L'uso dei colori

Dal momento che Amiga è una macchina con molti colori, come possiamo usarli?
Potrebbe essere interessante sapere verso quale angolo il punto si è diretto. Se associamo un colore (il che è facile su Amiga) ad ogni angolo, e coloriamo il punto con il colore dell'angolo verso cui stiamo andando, raggiungiamo lo scopo.

Che cosa accade? E' la prossima sorpresa. Ci aspetteremmo di vedere i tre colori mescolati caoticamente sullo schermo. E' esattamente quello che non avviene.
Il sorriso rimane, ma il gatto scompare.
Nella figura 2, il triangolo 1 viene riempito tutto con lo stesso colore: il colore A. Similmente, i pixel dell'area 2 sono tutti del colore B e quelli dell'area 3 del colore C.
Quando ci si pensa, la distribuzione del colore ha un senso. Si consideri che, in ogni caso, il prossimo punto starà a metà strada fra il punto corrente e un angolo. Ciò costringe il punto a situarsi a meno della metà della strada dall'angolo destinazione.
Ammesso che, per qualche ragione, il triangolo centrale sia irraggiungibile, i triangoli di Profondità = 0 sono colorati allo stesso modo del loro angolo. D'ora in poi questo lo chiameremo display a Profondità = 0.

Storia

Ora il display a Profondità = 0 sembra ovvio, almeno rispetto a prima. Il luogo in cui ci stiamo dirigendo sta risultando meno colorato di quanto sperassimo.
Dal momento che non possiamo prevedere la posizione futura dei punti, ci rimane solo il passato. Che accade quando coloriamo i punti con il colore dell'angolo verso cui ci stavamo dirigendo n turni fa? Un modo per modificare il loop per tenere traccia della storia è:

```
for (i=0; i < NumeroArbitrariamenteGrande ; i++)
{
  r = rand() % 3;
  for (j = StoriaMassima-1; j > 0; j--)
    Storia[j] = Storia[j-1]; /* sostituisce i
vecchi punti */
  Storia[0] = r;
  x = (x + Angolo[r].x)/2;
  y = (y + Angolo[r].y)/2;
  SetColour(ColoreAngolo(Storia[Profond]));
  SetPixel( rp, scala(x), scala(y));
}
```

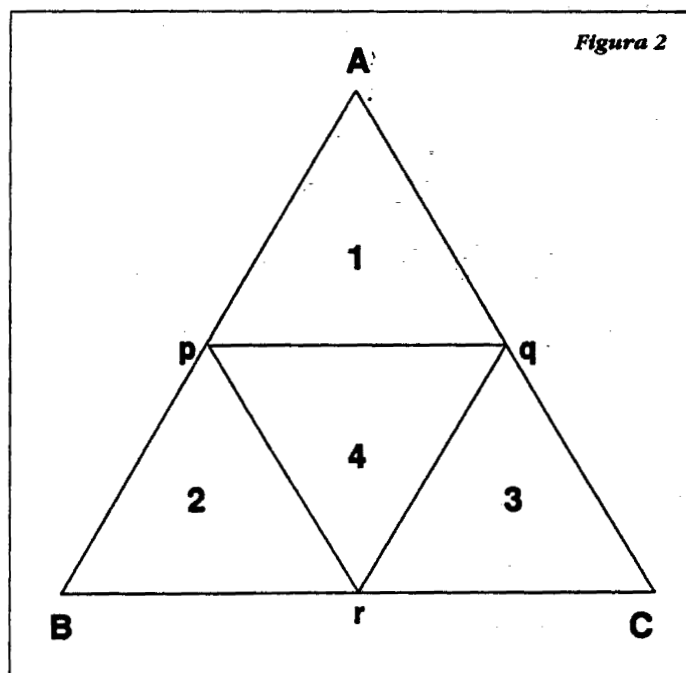
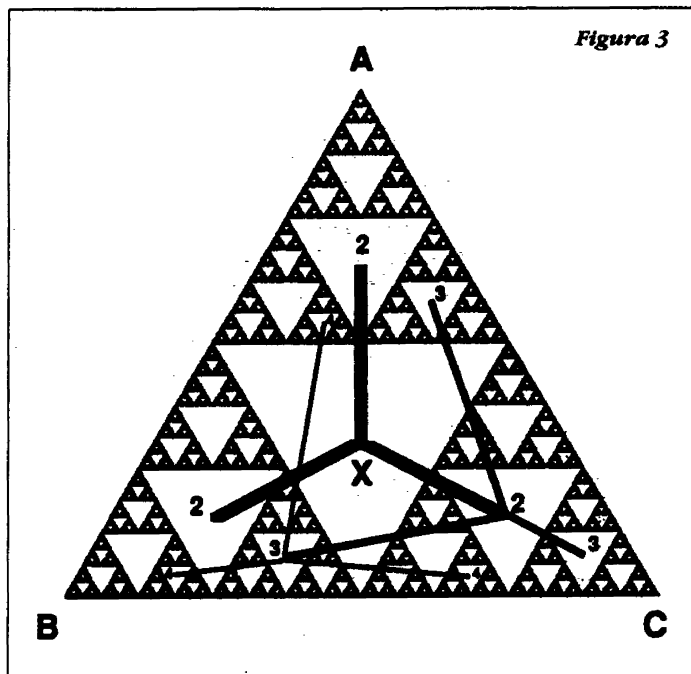



Figura 2

Sappiamo già cosa accade con Profondità = 0. Otterremo l'esplosione di un negozio di dolciumi con Profondità = 1? No, ciascuno dei triangoli 1, 2 e 3 in figura 2 è ora una copia dimezzata di quelli del display a Profondità = 0. La copia comprende persino i colori. Sia nell'1 che nel che nel 3, il colore A è ancora sulla cima, B a sinistra e C a destra. Terminiamo con 9 triangoli distinti monocolore.
E a Prondità = 2? Una trama comincia ad apparire... ogni triangolo monocolore di Profondità = 1 diventa una miniatura del display a Profondità = 0.

In generale, per una Profondità = n, si troveranno 3n miniature del display di Profondità = 0. Con un display di soli 400 x 400 pixel con cui giocare, questi triangoli in miniatura scompaiono attorno alla Profondità 6 o 7.
Ciò che questo implica è importante quanto profondo. Se si sa dove sta un punto, si può ripercorrere il sentiero che ha imboccato per giungere fino a lì! Ciò segue dal fatto che quando un punto si trova in particolare sotto-triangolo di Profondità = 0, esso deve essersi diretto verso l'angolo di quel sotto-triangolo.
L'algoritmo è semplice. Si raddoppi la distanza fra il punto e l'angolo più vicino: il punto precedente è sull'estremo, che si trova dalla parte opposta rispetto all'angolo, del segmento che passa attraverso l'angolo e il punto stesso.
Possiamo, naturalmente, ripetere questo algoritmo più e più volte finché...

Merita di essere analizzata l'azione dell'algoritmo di "retromarcia" quando il punto si trova al centro del triangolo (4 nella figura 2). Il punto precedente è al di fuori del triangolo. Non dovete tentare di farlo fare al programma!
Un'altra cosa da notare è che ogni punto nel triangolo sta a metà strada fra qualche punto al di fuori del triangolo e uno degli angoli. L'algoritmo di "retromarcia" non può determi-

nare dove è iniziata l'intera avventura. Il "percorso" che esso genera, assume, se possibile, che il punto precedente si trovava all'interno del triangolo.

Percorsi e alberi

Consideriamo, solo ai fini della discussione, di iniziare con un punto (segnato con una x) nel triangolo centrale di figura 3. Assumiamo di avere gettato un dado che ha fornito dei risultati tali che ci muoveremo in successione verso C, B e A. Abbiamo disegnato tutti i tre percorsi possibili a partire dal punto x verso i tre triangoli chiamati 2. Dal punto che si trova a metà strada verso C, abbiamo disegnato i tre percorsi possibili verso i triangoli chiamati 3 e così via.

Se noi avessimo disegnato tutti i possibili percorsi da ognuna delle destinazioni possibili, la figura 3 assomiglierebbe alla figura 1 quando questa fosse stata attaccata da un ragno malvagio. Si consideri, tuttavia, tale gruppo di linee. Ogni nodo avrebbe una linea in ingresso e tre in uscita. Tale insieme di linee e nodi è comunemente chiamato albero. E' un albero di grado 3. Cioè, ogni nodo può solo tornare verso il suo genitore (di qui la storia che abbiamo scoperto in precedenza) e ha tre figli, uno per ciascun numero casuale.
Sebbene l'attraversamento in avanti di un tale albero richieda un dado (o un generatore di numeri casuali), percorrerlo all'indietro è semplice. Ogni nodo ha un solo genitore. Abbiamo notato in precedenza che non è necessario che il percorso all'interno del triangolo cominci in un'area particolare. Possiamo mostrare che qualsiasi altro punto in cui il percorso si introduca nel triangolo sta su un albero che inizia nel triangolo centrale vuoto. Noi possiamo perciò chiamare gli alberi che cominciano in esso "Sequoie" (ehi, siamo della California, e così chiamiamo gli alberi grandi, da queste parti).

Quante Sequoie ci sono? Bene, c'è un infinito numero di punti nel triangolo centrale, e noi sappiamo che due alberi non condividono neanche un punto, dunque c'è un numero infinito di Sequoie.

Spazi vuoti

I colori ci hanno indicato la via per altre scoperte. Consideriamo tutti i punti posti a metà strada fra il segmento AC e il punto B (ci si riferisca alla figura 2). La linea formata è pr. Poi, si considerino tutti i punti lungo il segmento AB, poi tutti i punti che stanno a metà strada fra B e ognuno di loro. La linea così formata è pB.
Infine, troviamo che la linea rB deriva in maniera simile da BC. Perciò ogni punto nell'intero triangolo ABC, per esempio, che "si dirige" verso B finirà nel triangolo pBr (meglio noto come triangolo 2 in figura 2).

Siccome un triangolo è simmetrico, possiamo dedurre la stessa cosa per BC verso A e AB verso C. Questo significa che una volta ci si trovi entro uno dei triangoli a Profondità = 0, non si può più entrare in pqr. Si starà sempre all'interno del sotto-triangolo in cui ci si trova o si salterà oltre pqr.
Si noti che questo accade anche se veniamo dal triangolo pqr. Non appena usciamo da pqr verso uno degli angoli (il che accade in un solo turno), non torneremo più in pqr. Questo significa che se il nostro algoritmo di "retromarcia" genera un punto in pqr, abbiamo terminato. Il percorso deve essere iniziato qui.
Se applichiamo questa logica (guardando alla posizione dei punti posti a metà strada fra i nostri nuovi triangoli e ciascuno degli angoli), scopriamo che, una volta lasciato il grande spazio centrale al centro dei triangoli 1, 2 o 3, non possiamo più tornare indietro.

Area

"Ho visto spesso un gatto senza sorriso" pensò Alice, "ma un sorriso senza un gatto! E' la cosa più curiosa che ho visto in tutta la mia vita", Lewis Carrol, Alice nel paese delle meraviglie.
L'area della parte colorata del triangolo è 0. Si può vederlo in due modi differenti. La prima è pensare di avere un numero infinito di alberi e così vi troverete rapidamente trascinati nelle infinità di Aleph Null e Aleph One.
Fortunatamente c'è un metodo più semplice con cui calcolare l'area dello spazio vuoto. E' evidente che il triangolo vuoto più ampio è 1/4 dell'intero triangolo. Se li si esamina, i triangoli vuoti di Profondità = 1 sono 3/16 dell'intero triangolo. Il calcolo di qualche altro triangolo conduce alla formula:

$$\text{Area Vuota} = \lim_{n\to\infty} \frac{1}{4}\sum_{i=0}^{n}\frac{3^i}{4^i} \quad \text{o} \quad \frac{1}{4}\sum_{i=0}^{n}\left(\frac{3}{4}\right)^i$$

che è di fatto 1. Potete sia elaborare il limite o lanciare un

semplice programma per, diciamo, 10 iterazioni. Esso converge molto rapidamente verso 1.
Oppure, se ricordate che cosa ci è stato insegnato durante le lezioni di algebra:

si consideri la serie

```
S = r0 + r1 + r2... rn-1                    (1)
```

se moltiplichiamo entrambi i lati dell'equazione per r, otteniamo

```
rS = r1 + r2... rn                          (2)
```

se poi sottriaiamo (2) da (1) otteniamo:

```
S(1-r) = r0-rn                              (3)
```

Dal momento che r elevato 0 è 1, per tutti gli r (tranne infinito), l'equazione finale appare:

```
      1-rn
S = ------                                  (4)
      1-r
```

in questo caso abbiamo il caso di un limite con n tendente a infinito e siccome r è minore di 1, r elevato infinito è chiaramente 0. Cio ci lascia con:

```
S = 1/(1-r)                                 (5)
```

sostituendo 3/4 ad r si arriva a S = 4. Siccome vogliamo 1/4 di S, l'area in questione è 1.
Ciò che questo significa disturba. Se l'area vuota entro il triangolo è il 100%, l'area piena è lo zero per cento! Così, in realtà non c'è alcuna figura. La figura è un'illusione causata dal fatto che i pixel sono sensibilmente più grandi dei punti.

E' un frattale? E' Caos?

E' un frattale? Un frattale è una figura che ha infinite circonferenze in un'area finita. Ma ciò che abbiamo generato non è una figura, dal momento che i suoi punti non sono connessi. Inoltre, non ha area. Così non è un frattale.
E' caotica? I fenomeni caotici non sono ben definiti. Uno dei classici segni di un fenomeno caotico è la dipendenza dalle condizioni iniziali, con ciò i teorici si riferiscono al fatto che, se si cambiano leggermente i punti di partenza, si dovrebbero vedere differenze radicali.

Consideriamo il gioco del caos con differenti condizioni di partenza. Prendiamo mentalmente una coppia di punti (distanti quanto si voglia). Ok, gettiamo il dado.
Invece di segnare semplicemente il punto medio tra il nostro punto di partenza e l'angolo selezionato, pensiamo a questo.
Disegnamo delle linee che colleghino i tre punti (i due punti partenza e il punto finale). Ora segnamo i punti medi e disegnamo una linea tra di loro.

La figura 2 assomiglia a quella che abbiamo appena realizzato. Supponiamo che i punti B e C rappresentino i punti scelti inizialmente e A l'angolo del triangolo, allora p e q sarebbero i punti medi. Notiamo che abbiamo due triangoli simili (ABC e Apq).
Siccome sappiamo che Ap è la metà di AB, ne segue che pq è la metà di BC. Hmmm... dopo un turno i due punti dei nostri diversi percorsi distano solo della metà... e avanti così.

Dopo ogni "turno" del gioco del Caos possiamo disegnare mentalmente i triangoli. Scopriamo ancora un'altra coppia di triangoli simili, e i nuovi punti sul percorso distano solo della metà.

Possiamo continuare questo gioco del Caos, seguendo due percorsi separati, finché i punti sono arbitrariamente vicini fra loro. E, una volta vicini, continueranno a diventare sempre più vicini turno dopo turno. Questo non è sensibilità alle condizioni iniziali, questo è proprio l'opposto: dunque non è caotico!
Dunque che cos'è? E' pulito, elegante, non ovvio e ricorsivo. Esiste un'analogia fisica? Che ne direste di questa? E' un insieme casuale di campioni booleani, "sopra o sotto l'acqua", registrati ogni n secondi dopo che una pietra è stata gettata in uno stagno perfettamente triangolare.

Generare poligoni (n-goni) regolari

Data la nostra conoscenza dei triangoli, possiamo fare previsioni per altre figure abbastanza bene. Il problema è come generare gli angoli per figure regolari di n lati (n-goni)?
L'algoritmo qui presentato si basa sul fatto che in un n-gono regolare (con tutti i lati della stessa lunghezza), tutti i vertici giacciono su una circonferenza.
I passi poi sono:

- trovare il raggio del cerchio
- determinare il punto iniziale
- generare il vertice
- ripetere l'operazione lungo la circonferenza

Dobbiamo determinare il punto di partenza per costringere l'algoritmo a generare gli angoli dell'n-gono in modo che ci sia un vertice nel punto più basso.
Se il numero di lati è dispari:

```
r = AltezzaWindow/(1+cos(#/i))
```

Se il numero di lati è pari:

```
r = (AltezzaWindow/2)/(cos(#/i))
```

Anche le coordinate della base dipendono dal fatto che il numero di lati sia dispari:

```
if ( NumeroLati & 1 == 1) /* dispari ? */
{
  r = AltezzaWindow/(1+cos(pi/NumeroLati));
  cx = AltezzaWindow/2;
  cy = r;
}
else
{
  r = (AltezzaWindow/2)/(cos(pi/i);
  cx = AltezzaWindow/2;
  cy = AltezzaWindow/2;
}
```

Nel programma, sottriaiamo una piccola quantità ai lati per permettere il bordo. Ora dobbiamo generare i punti:

```
theta = (2*pi)/NumeroLati;
for (punto =0; punto < NumeroLati; punto++)
{
 Angolo[punto].x = r * sin (theta*((2*punto) 2)+cx);
 Angolo[punto].y= r * cos (theta*((2*punto)/2)+cy);
}
```

Più lati

Il programma dà all'utente una scelta fra n-goni con da 3 a 15 lati. In questo modo, una semplice finestra Amiga a 16 colori può visualizzare la storia usando i colori e ci rimane ancora un colore per lo sfondo.
Con più lati, tuttavia, vediamo generalmente un effetto "sale e pepe". Ciò è dovuto al fatto che continuiamo ad usare 1/2 della distanza fra l'ultimo punto e il prossimo selezionato.
Se aggiungiamo la possibilità di modificare questo rapporto, per esempio permettendo all'utente di fornire il denominatore e il numeratore, l'algoritmo cambia come segue:

```
r = rand() % NumeroLati;
/* aggiorna la storia come sopra */
x = (denom*x)/num+(num+Angolo[r].x)/denom;
y = (denom*y)/num+(num+Angolo[r].y)/denom;
```

Ciò significa che le nuove coordinate sono num/denum della distanza fra le vecchie coordinate e l'angolo scelto.
I nostri poligoni favoriti hanno NumeroLati pari a 7, 11, 12, 13 e un rapporto che costringa gli "anelli" a sovrapporsi leggermente.

Usare rand( )

Nel codice esempio, abbiamo usato fino ad ora la funzione standard C rand( ). Se tuttavia la usiamo con NumeroLati corrispondente a una potenza di due, otteniamo la ripetizione di un ciclo molto corto e pertanto pochi punti.
La Lattice fornisce lrand48( ), che non risolve questo problema. Dovemmo infine usare lrand48( ) e dividerlo prima di eseguire "% NumeroLati".
Forse un uso dell'algoritmo è come un altro test per generatori di numeri casuali.

Numeri in virgola mobile o interi?

Il programma usa interi (gli standard 32 bit della Lattice) per i punti e per quasi ogni altra cosa. Gli interi vengono scalati con un fattore di 10000, il che significa che dobbiamo dividere una coordinata per 10000 per ricavare la effettiva posizione del pixel.
Visto altrimenti, registriamo i punti come se stessimo disegnando su uno schermo di 4000000 x 4000000 pixel. Quando viene il momento di disegnarli su uno schermo 400 x 400, li dividiamo per 10000.
Sebbene possiate pensare che questo serva a incrementare la velocità, l'abbiamo fatto per un'altra ragione. Detto semplicemente, l'intero (o in realtà, il numero a "virgola fissa") è più accurato in questo caso.

Nello stesso numero di bit, il numero in virgola mobile deve coprire una "distanza" molto più grande, +/- 10 elevato 308 nel caso della Lattice.
Mentre è vero che molti dei punti compresi tra 0 e 399 (massima altezza dello schermo consentita dal programma) sono coperti dal normale funzionamento dei numeri in virgola mobile, noi ne ricaviamo molti di più, e con una differenza molto prevedibile (1/10000) con i numeri a "virgola fissa". Oh certo, sono anche più veloci.

Il programma

Il programma di cui abbiamo parlato si chiama Seth.
Dal momento che tutta la nostra grafica si inserisce perfettamente in un quadrato, abbiamo usato una finestra di 400 x 400 pixel. Seth usa una palette fissa di 15 colori più un colore di sfondo. Ciò limita il numero di lati a 15, ma se non si possono vedere le trame già con questi, dei lati in più non dovrebbero essere utili.
Il programma usa menu e gadget per permettere all'utente di controllare il numero di lati, la profondità della storia, il colore di sfondo e il rapporto per la divisione. Il programma continua ad eseguire il gioco finché non gli si dice di fare qualcosa di nuovo. Non abbiamo, di fatto, un NumeroArbitrariamenteGrande come costante di controllo del loop.

Il programma usa intensivamente il computer. Il solo I/O che effettua per lunghi periodi avviene verso lo schermo. Se volete far girare altri task sulla macchina, dovreste invocare questo programma con una bassa priorità.

Disponibilità del programma

Il programma si trova, naturalmente, sul disco che accompagna la rivista. E' stato scritto in Lattice C (5.0).
Si chiama Seth perché Seth è il dio egiziano del caos. Da quello che abbiamo visto, tuttavia, Horus, dio dell'ordine, potrebbe essere più appropriato.

sto di inserire il disco sorgente e premere il tasto destro del mouse. Alcuni dischi potranno non essere riconosciuti da AmigaDOS che risponderà con un System Message:

```
Error validating disk
Disk is unreadable/Disk structure
is corrupt
Use DISKDOCTOR to correct it
```

Ignoriamo questi messaggi e clicchiamo sempre su Cancel. Torniamo allo schermo di Boot Doctor per continuare il nostro lavoro tramite il gadget di profondità dello schermo o con la combinazione di tasti Amiga-sinistro+M. Dopo aver premuto il tasto destro del mouse, salviamo il nostro boot block su disco, tramite l'apposito file requester.
Nel caso il nostro gioco preferito venga infettato da un virus, tramite la settima opzione Restore Archival to Disk, sarà possibile riscrivere il boot block con la copia che ci eravamo precedentemente preparata. Dopo aver selezionato questa opzione, tramite l'apposito file requester. selezioniamo anche la copia di archivio del boot block e clicchiamo su OK. Dopo aver inserito il disco. diamo il via alla riscrittura del boot block, tramite il tasto destro del mouse. Ricordiamoci però di eliminare il virus anche sugli altri dischi, altrimenti ci vedremo costretti a ripetere questa operazione molte volte.
L'ottava opzione. Quit Boot Doctor. permette di abbandonare Boot Doctor. Come abbiamo accennato all'inizio, Boot Doctor non è un antivirus, ma un'utility per gestire i boot block. Prima di utilizzare qualsiasi opzione di Boot Doctor che scriva su disco. raccomandiamo di utilizzare l'opzione Save Archival BackUp of Disk per non rischiare di rovinare irrimediabilmente qualche gioco prezioso.

DSD

DSD, acronimo di Disk Storage Deluxe, è una piccola utility che mostra una lista dei device montati nel sistema, incluse anche le partizioni di un eventuale hard disk.
Una volta lanciato DSD, sia da CLI che da Workbench, appare una finestra che mostra le seguenti informazioni:
* Name: E' il nome AmigaDOS del device (ad esempio, DF0:,RAD ecc)
* Usage: Indica quanto è pieno il disco (in percentuale)
* Free: Mostra lo spazio libero su disco (in KiloByte)
* Size: Indica la massima capacità del disco (in KiloByte)
* Volume: Il nome del Volume (ad esempio, Amiga Magazine:)
* Alarm: E' il minimo numero di Kbyte che possono rimanere liberi in un device; superata questa soglia viene avvertito l'utente tramite quattro modi diversi selezionabili da menu.
* Chip: La quantità di memoria CHIP libera
* Fast: La quantità di memoria FAST libera
* Total: La quantità totale di memoria libera nel sistema (CHIP+FAST)
* Chunks: Il numero di blocchi contigui di memoria libera. Questo da un'idea della frammentazione della memoria.
* RAM Disk Size: La grandezza della RAM Disk (in KiloByte)

Nella finestra, inoltre, può comparire un asterisco " * " nel caso in cui un floppy disk è protetto da scrittura, oppure una " V " se il disco è in fase di " validate ". DSD può supportare un massimo di 20 device.
Disk Storage Deluxe possiede ben cinque menu a tendina: Options, Remove, Alarm, Save e Screens. Vediamoli brevemente:

Menu Options: In questo menu, si trovano una serie di opzioni per l'uso generale del programma.
-Reset All: Resetta la lista dei device
-Memory Display: Abilita il box che mostra la quantità di memoria disponibile.
-Time Display: Permette di mostrare, al posto del titolo della finestra, l'ora e la data. Può essere On o Off.
-RAM Disk Display: Abilita il box che mostra le informazioni sulla RAM Disk.
-Go To Sleep: Selezionando questa opzione, la finestra viene rimpicciolita e il programma rimane "congelato" (in lista di attesa). Cliccando dentro questa piccola finestra, il programma riprende normalmente.
-Pop up to front: Quando viene selezionata questa opzione, la finestra di DSD viene sempre portata sopra tutte le altre, su qualsiasi schermo ci troviamo.
-Quit: Abbandona il programma

Menu Remove: Tramite le opzioni di questo menu, è possibile selezionare, seguendo alcuni criteri, quali device devono essere mostrati.
-Selected: Rimuove dalla lista dei device tutti i device selezionati (per selezionare un device basta cliccare sulla sua riga).
-Unselected: Rimuove tutti i device non selezionati.
-Floppies: Rimuove tutti i device che iniziano per " DF ".
-Not Floppies: Rimuove tutti i device che non iniziano che " DF ".
-< Alarm Level: Rimuove tutti i device che hanno uno spazio su disco minore del livello di allarme.
-> Alarm Level: Rimuove tutti i device che hanno uno spazio su disco maggiore del livello di allarme.
-Empty Drives: Rimuove dalla lista dei device tutti quelli che non hanno il disco inserito.
-Everything: Toglie tutti i device dalla lista.

Menu Alarm: Questo menu permette di scegliere quale tipo di avviso DSD deve dare all'utente, quando lo spazio su disco passa il livello di allarme.
-None: non viene effettuato nessun allarme.
-Flash: viene fatto lampeggiare lo schermo 5 volte e contemporaneamente lampeggia anche la riga del device che ha caudato l'allarme.
-Requester: viene aperto un requester su qualsiasi schermo.
-Speech: il programma vi informa dell'accaduto (nella directory LIBS: deve essere però presente translator.library)
-Beep: vengono fatte tre segnalazioni acustiche.

Menu Save: Tramite questo menu, è

possibile salvare la configurazione di DSD e la lista dei device.
-Config: Salva alcuni parametri del programma, come le opzioni del menu Options, Remove e Alarm, oltre alla posizione della finestra e i livelli di allarme per la RAM Disk, la memoria CHIP e la memoria FAST. Il file di configurazione viene salvato con il nome di "DSD-Config" nella directory S: e viene automaticamente caricato la volta successiva.
-Devices: Salva la lista dei device con il loro rispettivo livello di allarme nel file " DSD-Devices" nella directory S:. Il file verrà poi caricato automaticamente la prossima volta che DSD sarà mandato in esecuzione.

Menu Screens: Tramite questo menu, è possibile selezionare su quale schermo fare apparire DSD. In questo menu è infatti disponibile (fino ad un massimo di 10) la lista degli schermi presenti in quel momento. Dopo aver selezionato lo schermo desiderato, quest'ultimo viene portato in primo piano insieme a DSD. Vi sono, inoltre, alcuni gadget nascosti che hanno alcuni funzioni interessanti. Ad esempio, cliccando su Name: vengono deselezionati tutti i device; cliccando invece su Used viene mostrato Size e viceversa. Per mostrare Volume, si deve invece cliccare su Free o Alarm. Il gadget di chiusura non chiude totalmente la finestra e fa finire il programma come molti si potevano aspettare, ma riduce soltanto la grandezza della finestra alla barra del titolo, che potrà mostrare il titolo della finestra o l'ora e la data, a seconda della selezione. Cliccando nel box della memoria sulle prime due colonne, si cambia il colore della finestra, mentre cliccando sull'ultima colonna (TOTAL) appare CHUNKS.Per selezionare il livello di allarme per ogni device, basta cliccare sulla riga del device, mentre per impostare il livello di allarme della memoria CHIPo FAST,bisogna cliccare nel boxRAM Disk Size, ripetutamente, sino a che non compare quella desiderata. Per ulteriori informazioni si consiglia di leggere il file DSD.DOC nella directory Utilities.

• Cli Utility

In questa directory sono contenute, in diverse sub-directory, alcune piccole utility che possono rendersi utili nel lavoro di tutti i giorni. Queste utility sono utilizzabili solo da CLI e sono particolarmente interessanti per la presenza del file sorgente in alcune di esse.

Filter

Filter legge un file in input e mostra il contenuto del file a video, utilizzando solo ed esclusivamente codice ASCII (da 0 a 127). Incluso il file sorgente in linguaggio C.

Ins_Returns

Ins_Returns legge un file di input e scrive l'output inserendo, ad intervalli appropriati, dei Carriage Return. Questa utility si rivela interessante quando si deve modificare un file di testo con un editor di linea (tipo Edit). Anche in questa directory è presente il file sorgente in C.

PWFix

Corregge le strutture create con le vecchie versioni di Power Window che il compilatore Manx non produce dei messaggi di avvertimento (compiler warning). Incluso il sorgente.

SetPrompt

SetPrompt permette di utilizzare nel prompt del CLI i colori e vari stili (grassetto, corsivo e sottolineato). E' l'unico programma a non avere il sorgente.

Split

Split è forse il programma più utile di tutta la collezione. Esso permette di spezzettare file di grosse dimensioni in più file, in modo da essere portato su più dischi ed essere poi ricomposto con il comando Join dell'AmigaDOS.

StripBin

Legge un file ed elimina i caratteri non-ASCII.L'output può essere editato con ED. E' incluso il sorgente.

Tree

E', per certi versi, simile a "dir opt a", ma non stampa i file; viene mostrata solo la struttura della directory.

XChar

Elimina i caratteri non ASCII da un file di input.

• Games

EgyptianRun

E' una divertente corsa nel deserto. Il gioco necessita del joystick nella porta 2. Si tratta di fare un rally tra le piramidi e arrivare fino in fondo. Si parte a 50 Km dalla meta e, via via, il percorso si fa più veloce e insidioso. Se colpiamo una piramide, bisogna ricominciare tutto da capo, ma prima, un 'simpatico' teschio ci mostra una delle sue risate meglio riuscite. Fate attenzione anche alle pozze d'acqua, sulle quali la mostra macchinina scivola facilmente. Buon divertimento !

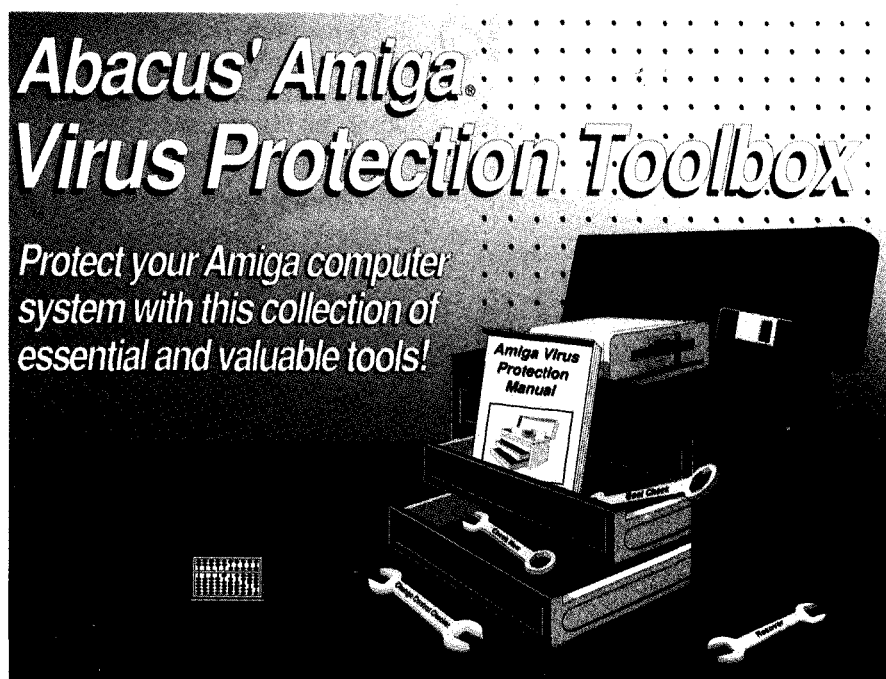
Virus Protection Toolbox

Il pacchetto anti-virus più completo.

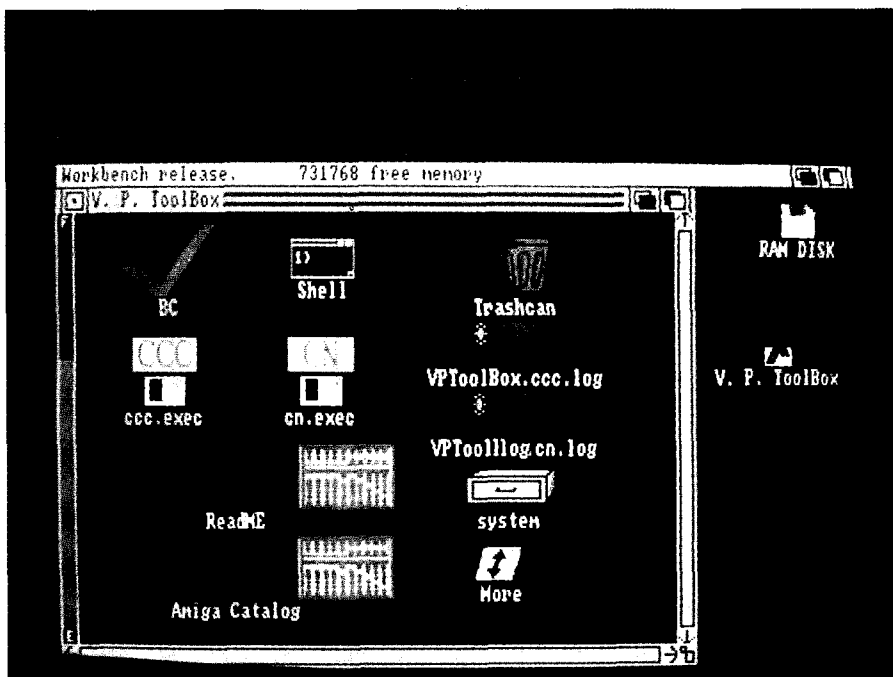
Marco Tortolina

Negli ultimi tempi i problemi creati dai "virus" informatici hanno avuto una notevole risonanza nell'informazione quotidiana propinataci dai grandi mass media. Le notizie che ci sono arrivate non hanno però fatto altro che creare una grande confusione; infatti se solo qualche anno fa l'idea che una specie di "virus" avrebbe potuto infettare la memoria del nostro computer, con i relativi programmi, faceva sorridere, oggi serpeggia un panico non del tutto giustificato e il pensiero che qualche orribile organismo vorace di silicio si aggiri tra i chip del nostro calcolatore, rende sicuramente poco tranquilli molti utenti. Nulla di tutto questo: i "virus" informatici sono programmi e basta! Proprio come un database o un word processor, magari anche più ingegnosi e complessi, ma sostanzialmente programmi sono e programmi rimangono e come tali possono essere individuati ed eliminati da altri programmi: gli antivirus.

Toolbox

La ABACUS, partendo da questo presupposto, ha creato VIRUS PROTECTION TOOLBOX, un pacchetto che si inserisce tra i più completi del suo genere. Fortunatamente per gli utenti di Amiga non si può parlare finora di virus particolarmente dannosi; la loro individuazione richiede l'uso di semplici analizzatori del BOOT BLOCK e il solo IRQ, peraltro abbastanza innocuo, contagia programmi del disco. Il Virus Protection TOOLBOX vuole

essere, sulla base dell'esperienza di altri sistemi, un'anticipazione allo sviluppo che i virus di Amiga potranno avere in un prossimo futuro (Trojan Horse, Worm, ecc.).
Virus Protection TOOLBOX è un insieme di quattro programmi contenuti su un disco, accompagnati da un buon manuale, in inglese, che spiega anche esaurientemente il funzionamento dei vari tipi di virus di Amiga finora conosciuti.
Vediamo ora in dettaglio il funzionamento e le caratteristiche dei programmi.

B.C. (Boot Check)

B.C. può partire sia da Workbench, con il classico doppio click sulla sua icona, che da CLI o Shell, però per utilizzarlo al meglio è consigliabile seguire attentamente quanto segue.
Preparato un backup del nostro Workbench, sul quale avremo trasferito i programmi di Virus Protection TOOLBOX, potremmo attivare B.C. con il comando RUN BC inserito nella startup-sequence; una volta caricato risiederà in memoria controllando che le locazioni e i vettori delle librerie, normalmente visitate da virus, siano in ordine; nel caso contrario ci viene segnalata la presenza e il tipo di virus (B.C. riconosce: SCA, Byte Bandit e Byte Warrior, i tre principali da cui sono derivati i relativi cloni) che sarà quindi eliminato riportando alla normalità le funzioni del calcolatore. Inoltre, B.C. controllerà ogni disco che inseriremo, d'ora in avanti, nei drive e trovando un Boot Block infettato o non standard un "System-Alarm" (tipo guru) ce lo segnala e ci chiederà se vorremo INSTALLare il disco o meno.
Accettare questa operazione comporta la sostituzione dei settori del boot incriminato con settori standard.
Ciononostante, non avremo perso il vecchio boot, infatti questo è conservato ancora in memoria ed è pronto, se si vorrà, ad essere salvato in un log.file su RAM disk. Da notare che se il disco in oggetto è protetto in scrittura, l'INSTALL non avrà naturalmente successo, ma il boot potrà venire ugualmente salvato. Prima di resettare sarà utile trasferire il log.file (magari cambiandogli nome) su un disco dati.
Con tutto questo è quindi stato possibile formare un vero e proprio "database" per i Boot Block particolari di cui sono dotati molti dischi (specialmente i game) che richiamano appositi caricatori senza i quali il programma sarebbe inutilizzabile.

ReadME
The Amiga Virus Protection ToolBox contains the following programs:

BC (BootCheck) Checks the boot block of each disk
CCC (Change Control Checker)
This program is used to test for changes in important files on your disks
CN (CheckNew) Checks for new files on a disk
RECOVER This program is used to recover boot sectors save by the BootCheck program
VPToolBox.log This file is used by the Change Control Checker program to verify the contents Of the Virus Protection ToolBox disk.

Tips on Using the Abacus Amiga Virus Protection ToolBox Disk.

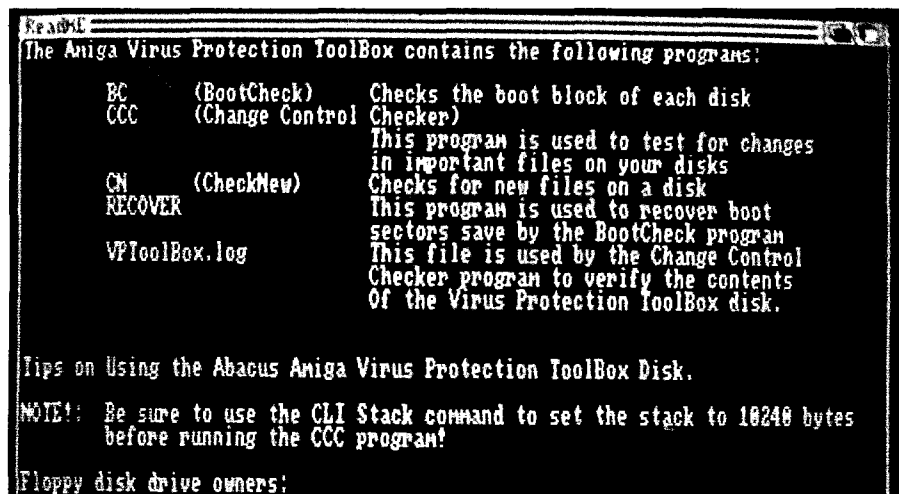
NOTE!: Be sure to use the CLI Stack command to set the stack to 10240 bytes before running the CCC program!

Floppy disk drive owners:

Recover

Lanciato da CLI (sintassi: RECOVER DRIVE boot block.file) è in pratica un appendice di Boot Check di cui sfrutta l'archivio sul quale si è catalogato i boot block che più ci stavano a cuore.
La funzione di RECOVER è, infatti, quella di rimpiazzare i settori di un boot eventualmente danneggiato o infettato con i proprii originali precedentemente catalogati.

CCC (Change control checker)

Si attiva anch'esso da WB o da CLI

come B.C. e necessita di uno STACK di 10240 per funzionare al meglio.
Il suo compito è quello di passare in rassegna alcuni o tutti (dipende dai parametri opzionali che si vorranno usare) i file di cui è composto il disco che stiamo esaminando.
Tramite il parametro "-o", che non è opzionale, stabiliremo dove si vorrà salvare l'output. Per esempio, per creare un logfile sul quale archiviare tutti i file del disco in DF0:, occorre far partire CCC in questo modo:

CCC df0:#? -odiskname.

Il parametro "-o" puo essere usato anche per assegnare l'input da un device, ad esempio:

CCC -oram:logfile

leggerà l'archivio da RAM disk. Il sopra citato archivio (logfile) sarà richiamato ogni qualvolta si vorrà accertare che il contenuto dei dischi non abbia subito alterazioni.
Lo scopo del programma è quello di scoprire, confrontando i file ai vecchi precedentemente analizzati, eventuali LINK virus, anticipando così azioni deleterie.
Solitamente questi virus modificano la struttura e la dimensione del file a cui si sono "aggrappati" e pur lasciandolo normalmente eseguibile (esempio l'IRQ virus), costituiscono una minaccia a tutto il nostro software. Se un file verrà trovato e modificato in qualsiasi sua parte, un "Brief-Message" ci darà il modo di vedere in che maniera e momento è stato alterato. Aggiungendo dei parametri (-a -b -d -e -f -i -p -n) al comando di lancio del programma, si può scegliere quali file si vorranno esaminare, per esempio con:

ccc df0:logfile -p

saranno passati solo i file eseguibili che si trovano sul drive df0, mentre con:

ccc ram:logfile -<d15-04-90

si controlleranno quelli con data antecedente il 15 marzo 1990 che si trovano su RAM disk e così via.
La consultazione del manuale è comunque indispensabile data la grande versatilità che i parametri conferiscono a CCC.

CN (Check new)

Il programma è simile a CCC, stesso caricamento e stessa sintassi, da cui si differenzia per le diverse funzioni che svolgono i parametri opzionali (-a -b -d -i -l -n -s) e naturalmente per lo scopo. Infatti, mentre CCC controlla la costituzione dei file, CN si limita a formarne uno schedario dal quale in seguito trarrà paragone al fine di identificare file che si sono aggiunti o che sono stati cancellati.
Questo potrà essere un campanello d'allarme nel caso in cui le modifiche non sono state volute ma sono state eseguite automaticamente a nostra insaputa.

Note

I programmi sono sfruttabili anche dai possessori di hard disk che avranno, anzi, il vantaggio di poterli inserire sul disco fisso avendoli perciò sempre e istantaneamente disponibili per i controlli di rito.
Virus Protection TOOLBOX è sicuramente una valida risposta ai fantasiosi creatori di virus e anche se il suo impiego può sembrare un po' macchinoso, usandolo con metodo risulterà molto utile al fine di evitare sgraditi contrattempi.
E' infine doveroso ricordare che un uso non corretto di B.C. e RECOVER potrebbe causare la perdita accidentale di alcuni programmi (molti hanno la protezione proprio nel boot block e senza questa sarebbero inutilizzabili).
Meglio, quindi, operare quando si è assolutamente certi di quello che si sta facendo. ▲

ABACUS
5370 5nd Street SE,
Gran Rapids, HI 49512
U.S.A.

Previsioni e speranze

Nuove curiosità dal meraviglioso mondo Amiga

Sebastiano Vigna

Ed ecco che finalmente si può parlare della 2.0 (ex 1.4), finalmente siamo sicuri che la Commodore non ha intenzione di mollare l'osso, finalmente tante speranze sono state realizzate.
E ora? Ora siamo un gruppo relativamente ristretto (la base installata di compatibili IBM è più di otto volte quella Amiga, e la maggior parte di quest'ultima è costituita da 500 usati come videogames) alla ricerca di un'identità e di un mercato. Alla ricerca, tanto per essere chiari, di un modo per non morire.
Pessimista? No, anzi. Gli ottimisti pensano di sapere come vanno le cose, i pessimisti lo sanno. Dunque vediamo un po' di fare il punto della situazione.
Il nuovo modello che verrà introdotto dalla Commodore e il nuovo sistema operativo dovrebbero essere un vero e proprio rilancio della serie Amiga.
Già, ora abbiamo una nicchia nel mercato desktop video ma, attenzione!, il Mac ce la sta rubando, perché sebbene costi ben più del doppio, ha anche ben più del doppio dei colori. E ben più del doppio del software disponibile. E costa, comunque, molto meno di quanto siano abituati a spendere gli acquirenti delle videostazioni per la produzione e la postproduzione di video.
Come contrastare questo andamento? Dove inserirsi? Da questo punto di vista la Commodore (o meglio, il nuovo, aggressivo staff dirigenziale della Commodore) ha elaborato alcune linee di attacco estremamente valide, di cui penso sia giusto parlare per bilanciare le mie invettive dello scorso numero:

1) Unix 4.1: con il port del sistema operativo su cui oggi sono puntati gli occhi di tutti, Amiga diventa una macchina Unix a basso costo estremamente appetibile. Esiste anche un programma, sviluppato in Germania, che trasforma un Amiga in un efficentissimo terminale X-Windows, aprendo così le porte all'utilizzo della macchina come terminale intelligente (e il settore dei terminali X-Windows è in rapida ascesa).
Il punto dolente che, purtroppo, non si può fare a meno di toccare, è che in questo modo l'originalità di Amiga viene sostanzialmente uccisa: abbiamo una macchina Unix ben funzionante ed economica, ma non abbiamo più Exec, né Intuition, né il software a cui ormai siamo affezionati, come CED, TxEd, AmigaTeX...

2) La Commodore sta portando avanti una politica di "promozione" di Amiga presso colossi del software in ambiente IBM, spingendoli (soprattutto con l'arrivo della 2.0, che rende molto più facile l'ingegnerizzazione del software) a trasportare i loro prodotti su Amiga.
Si rumoreggia, badate bene, si rumoreggia, che FoxBase e Lotus 1-2-3 siano sulla lista.
Che sia vero o no, è un dato di fatto che l'ingresso delle "majors" informatiche non può che fare affluire nuove energie, vedasi, ad esempio, WordPerfect, le cui impressionanti capacità hanno stimolato la concorrenza "Amiga-only" a produrre nuovi, potenti prodotti come excellence!. Chi può prevedere quale incentivo per l'acquisto sarebbe il sapere che Excel, tanto per dire, è disponibile su Amiga ?

(segue a pag. 58)

Post 1.1 & VLT

Altri due favolosi programmi di Public Domain

Sebastiano Vigna

Come ho già detto tempo fa, introducendo questa serie di articoli, spesso in ambito PD è possibile reperire programmi di altissimo livello, alla pari di prodotti commerciali o addirittura privi di una tale controparte.
Tale è, ad esempio, il caso di Post, un meraviglioso, polivalente interprete PostScript scritto da Adrian Alyward. Il programma è (come ormai diventa sempre più frequentemente) liberamente distribuibile, sebbene l'autore mantenga il copyright.
Il PostScript, sviluppato dalla Adobe, è un linguaggio orientato alla grafica che permette di descrivere, tramite un file ASCII, un output grafico in maniera indipendente sia dalla risoluzione disponibile, che dalla periferica usata. E' estremamente diffuso su stampanti laser di ogni marca, e si è ormai imposto come standard tra i linguaggi descrittori di pagina.
Normalmente, un file Postscript viene "digerito" dall'interprete contenuto nella stampante, che provvede a generare un'opportuna bitmap.
Le royalty che la Adobe impone per l'implementazione di un tale interprete sono estremamente alte, per cui le stampanti Postscript sono tipicamente costose.
I fatto di disporre direttamente su Amiga di un interprete che analizza un file, lo trasforma in una bitmap e poi lo manda ad una qualsiasi stampante accessibile attraverso le preferences, non è evidentemente cosa da poco.
A tutto questo, Post aggiunge un'incredibile flessibilità dovuta alle molteplici possibilità di output disponibili: è possibile visualizzare il file su uno schermo ad alta risoluzione o scriverlo su un file IFF ILBM, in modo da fare un preview. E' addirittura possibile fare funzionare l'interprete in modo interattivo, inserendo una riga di comandi per volta e vedendone in tempo reale, sullo schermo, l'effetto.
L'unico problema riscontrabile nell'uso, è il fatto che la Adobe ha il copyright sui font (Times - Roman, Helvetica etc.), per cui non è possibile utilizzarli con Post.
Bisogna procurarsi dei font di pubblico dominio oppure utilizzare solo le funzioni grafiche del Postscript. Un uso possibile è anche quello di "intermediario" per AmigaTeX. Se, ad esempio, non si possiede un driver TeX per Epson LQ ma si possiede quello per il PostScript, è possibile stampare il tutto prima in formato PostScript e poi lasciare a Post il compito di creare una bitmap adatta alla Epson.
Le possibilità sono pressoché illimitate, e non si può che ringraziare Adrian Alyward per il magnifico lavoro fatto: implementare un interprete di questo genere non è esattamente una cosa alla portata di chiunque.

VLT

Programma pochissimo noto in Italia, VLT meriterebbe una ben maggiore attenzione: si tratta di un programma di telecomunicazione sui generis.
E' stato sviluppato da William Langeveld allo SLAC (Stanford Linear Accelerator Center) ove un grande numero di Amiga è costantemente al lavoro sia per comporre documenti (e ci pensano TxEd e Amiga-

TeX), che per comunicare con il *mainframe* principale.
Per quest'ultimo scopo, trovando insoddisfacenti i programmi disponibili, sulla base del vecchio VT100 è stato sviluppato un potentissimo programma che però risente (inevitabilmente) del fatto di essere nato in un ambiente molto specifico. Tanto per essere più chiari, VLT non possiede alcuna funzione del tipo agenda telefonica con composizione automatica, né memorizzazione automatica di password. Anche il numero di protocolli disponibili è relativamente limitato. Cos'è, allora, che lo rende così attraente?
Punto primo: VLT è solido come una roccia. E' un programma utilizzato attivamente, e altrettanto attivamente supportato e aggiornato. Inoltre, è intrinsecamente "scritto bene", nel senso che si "fonde" perfettamente con il sistema operativo di Amiga e ne sfrutta tutti i lati migliori.
Punto secondo: a parte quello che riguarda l'entrata in connessione con la rete telefonica, VLT non è secondo a nessuno.
La velocità del suo protocollo ZModem è eccezionale, e il protocollo stesso è implementato tramite una libreria condivisa, a cui anche altri programmi possono accedere, secondo uno standard che lo stesso Langeveld ha ideato e che sta prendendo piede tra i programmi di comunicazione per Amiga. Sono disponibili anche XModem (128/1024 byte) e Kermit.
Esiste ovviamente un'opzione per catturare o revisionare il testo in arrivo, anche in tempo reale. I tasti funzione sono programmabili e lo schermo può essere in uno qualunque dei modi grafici di Amiga (purché ad alta risoluzione).
Un'altra chicca è l'interfaccia ARexx, che consente di mettere in collegamento VLT con altri programmi, o di scriversi con relativa facilità un programma di memorizzazione dei numeri telefonici più usati.
Non manca, infine, un emulatore completo Tektronix, per i fanatici di grafica. Non ho mai trovato un'occasione in cui potesse essere utile, ma non si sa mai.
Ci sono dozzine di altre caratteristiche uniche in VLT, ma credo a questo punto di aver dato un'idea della vastità del programma.
L'unica pecca è evidentemente la mancanza di funzioni di supporto per l'aggancio alla rete telefonica. Se potete passare sopra a questo problema, VLT è *il* programma di telecomunicazione. ▲

(segue da pag. 56)

Previsioni e speranze

Lo so, di primo acchito l'Amigoide medio storce il naso e sibila "e chi ha bisogno di questi?", ma ogni tanto una sano test di realtà può essere utile.

3) In particolare negli USA, una politica commerciale estremamente aggressiva sta portando Amiga sulla bocca di tutti. Byte dedica una copertina al 3000. Centinaia di spot su tutte le reti. E, questa è nuova, un servizio postale razzo gratuito tramite la Federal Express per accelerare i tempi di riparazione in garanzia. La Commodore sta cercando di farsi benvolere in casa sua (ci sono più Amiga venduti in Germania che in USA, e questo non è bene...), e forse ci sta riuscendo.

4) Una serie impressionante di schede per la messa in rete sta mettendo Amiga in grado di collaborare con le più disparate macchine, sia tra i mini che tra i micro, ponendo fine all'immagine "solitaria" che hanno sempre avuto i personal Commodore, viventi apparentemente in un mondo tutto loro. Non è più necessario pensare di avere per forza dieci Amiga, e solo quelli: si può anche aggiungere un Amiga o due ad una rete con VAX, Macintosh e IBM, ed utilizzarlo per i compiti nei quali è insuperabile (preview di documenti TeX, postproduzione, grafica in tempo reale...).

5) Gli sviluppatori di software vengono supportati, documentati, informati. I volumi di riferimento tecnico sono stati aggiornati e ristampati. C'è una rivista pubblicata dal CATS (Commodore-Amiga Technical Service) che parla dei misteri più profondi di Amiga, descritti dai suoi creatori o comunque da persone dotate di una profonda esperienza. Ci sono congressi organizzati dalla Commodore per mettere periodicamente al corrente delle novità gli sviluppatori (i famosi DevCon). Insomma, la Commodore sta anche cercando di farsi benvolere da chi scrive software per Amiga.

Conclusione: Amiga sta provando a decollare. Se il tentativo avrà successo, lo sapremo solo fra parecchi mesi, e nel frattempo non ci resta che incrociare le dita e fargli la migliore pubblicità possibile.
Perché ci dispiacerebbe vederlo soccombere, no? ▲

Expoedit Copia: mille novità e Amiga al top

Nel campo del DTP viene insediata la posizione privilegiata di Apple. Accanto a nuove macchine e workstation dalle prestazioni rivoluzionarie, il famoso Amiga 3000 con due nuove e sorprendenti release.

Luisella Acquati

Foto di Carlo Cattoni

La nuova edizione di Expoedit Copia, l'unica fiera italiana dedicata al Desktop Publishing, ai sistemi di stampa e fotocomposizione, è stata la vetrina ideale per la nuova proposta professionale di Commodore, Amiga 3000, dedicata ai grafici più esigenti!

Alla manifestazione, che ha raccolto la presenza qualificata di 60 espositori, erano presenti le più prestigiose marche di computer e periferiche anche se Apple, con il suo mega-stand, faceva la parte del leone. Infatti, le macchine Mac non riempivano solo lo stand Apple, a conferma del fatto che l'equazione DTP=MAC è ancora molto forte.

Ma non sono mancate anche proposte targate Apollo (una workstation con sistema operativo Unix) e Olivetti M290.

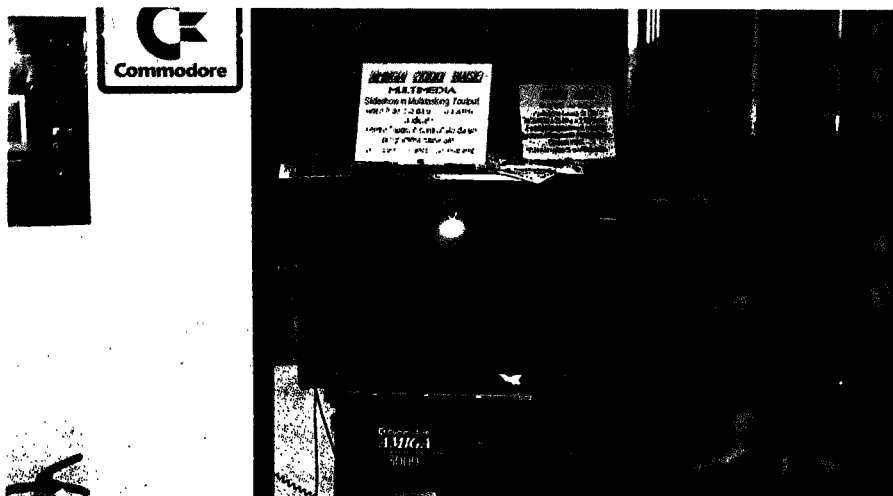
Olivetti Systems & Network ha puntato sulla stampa di documenti a colori, con la nuovissima stampante Tektronix a trasferimento termico con interprete Postscript, con dimostrazioni in Page Maker.

Particolarmente interessante è risultato il nuovo scanner a colori Microtek con risoluzione 300 dpi, 16.7 milioni di colori, area di scansione A3, presentato da Telcom e collegato a un Mac con scheda 32 bit.

Per chi ha necessità di archiviazione di immagini Maxtor ha presentato un disco ottico, da 1,2 Gigabyte riscrivibile e removibile, collegato a un PC.

IBM, con una presenza d'immagine, ha proposto come unica novità una postazione dedicata al Multimedia anche se, apparentemente, sembrava una semplice slide show.

Elcom ha presentato, per questa occasione, una simpatica animazione genlockata con una telecamera e un software di modellazione che ha attirato l'interesse di videografici e tecnici.
Lo stand Apple ospitava prodotti di aziende ad essa collegate come i nuovi monitor di Modo, i particolari Radius capaci di emulare a video i colori Pantone e l'ultimissima relase di Delta. Agfa, Linotronic, VariTyper e Offset hanno esposto i loro modelli più recenti dotati di massima precisione e maggiore velocità. Archie, invece, ha presentato Archimedes R140 in configurazione DTP, leggermente sottotono per quanto riguarda la velocità di risposta.
Ma sicuramente, la vera novità di questa fiera è stata la proposta Amiga 3000, una proposta DTP presentata pubblicamente per la prima volta da Digimail, azienda milanese che offriva quattro postazioni Amiga, dalla più economica basata su un Amiga 500 (equipaggiata con hard disk A590, 3 Mb RAM, monitor Commodore A2024) alla più sofisticata Amiga 3000 con stampante laser Postscript.
Digimail ha allestito per i visitatori un ambiente Multimedia con un gigantesco monitor Barco da 33 pollici stereo collegato ad un Amiga 2000 dove girava uno slide show costruito con De Luxe Video III pieno di immagini, animazioni e musica di sottofondo.
L'attenzione dei visitatori era molto spesso attirata più dal mega monitor che dai programmi di impaginazione Amiga anche perché era l'unica applicazione del genere !
Un Amiga 2500 con scheda acceleratrice 2630 e monitor 19" Viking 1, dimostrava le indiscusse qualità del Professional Page, soprattutto nella precisa rappresentazione a video dei font e dell'uscita in quadricromia. Amiga 3000 ha potuto invece dimostrare che Pagestream può essere perfettamente concorrenziale a Xpress o Page Maker. Non ha mancato di stupire il nuovo Workbench 2.0 che apriva uno screen di 2500 x 1500 punti su un normale Multisync.
Uno scanner Sharp JX300 capace di leggere un'immagine in 16.7 milioni di colori era collegato a un Amiga 2000 tramite scheda IEE488 di ASDG e i risultati erano di ottima qualità anche nei soliti 4096 colori.
Per gli amanti di MS-DOS e Mac, i dimostratori di Amiga hanno mostrato numerose applicazioni di programmi quali Caligari, Turbo Silver, Animagic o Sculpt 4D suscitando ammirazione sia per i risultati, che per i costi estremamente contenuti del software.
Particolarmente interessante è stata la dimostrazione del nuovo Vectrotrace, un programma che trasforma un disegno bitmap nel corrispondente vettoriale, non ancora disponibile in Italia ma che verrà distribuito, probabilmente, da Digimail.
Il risultato della trasformazione può essere salvato in vari formati tra cui Pro Draw Clip, Aegis Draw e Videoscape.
Poche ore prima della chiusura della fiera un interessante fuori programma: due innovative relase, denominate Professional Draw 2.0 e Pagestream 2.0, hanno mostrato caratteristiche rivoluzionarie: Pro Draw è circa cinque volte più veloce e Pagestream adotta ufficialmente i font Compugraphic.

Concludendo, va quindi sottolineato che Amiga sta scalando le vette del mercato professionale con progetti innovativi e di qualità come ha dimostrato a Expoedit '90.
A questo punto non resta che attendere la conferma di questa tesi nei prossimi appuntamenti d'autunno. ▲

Come utilizzare il dischetto

Prima di utilizzare il dischetto allegato alla rivista assicuratevi di averlo protetto dalla scrittura aprendone la finestrella, per evitare accidentali cancellazioni o contaminazioni da virus (il disco allegato è al 100% privo di virus ed il Gruppo Editoriale Jackson non si assume nessuna responsabilità in caso di contagio). A questo punto accendete il computer. Dopo la presentazione, dalla quale è possibile uscire premendo il tasto sinistro del mouse, verrà caricato il Workbench. Aprendo l'icona del disco (cliccando due volte con il tasto sinistro su di esso) appaiono i "cassetti" relativi a giochi, utilities, ecc. Aprite allo stesso modo il "cassetto" al quale siete interessati e caricate il programma desiderato. Alcuni programmi possono essere anche caricato da CLI con altre opzioni (vedere a parte le spiegazioni dei programmi). Per chi possiede Amiga con "soli" 512 K di memoria RAM è consigliabile chiudere tutte le finestre aperte tranne quella relativa al programma da far girare (che può essere eventualmente chiusa con un po' di prontezza di riflessi durante il caricamento). In tal modo il programma avrà a disposizione più memoria. In caso contrario, potrebbero verificarsi dei malfunzionamenti.

ATTENZIONE: prima di far partire ogni programma vi raccomandiamo di leggere attentamente le istruzioni del programma. I programmi DSD, Filter, Pw Fix, Split, StripBin, Tree ed Egyptian Run sono di pubblico dominio, e possono essere liberamente distribuiti. Inoltre, ricordiamo che, i programmi sorgenti ed eseguibili delle pagine di TRANSACTOR per AMIGA, si trovano nel cassetto TRANSACTOR e devono essere caricati da CLI. Tutti gli altri programmi presenti sul dischetto sono, invece, coperti da Copyright 1990 Gruppo Editoriale Jackson-Compute! Publications Inc., e come tali non devono essere copiati e distribuiti senza la preventiva autorizzazione scritta dell'editore. Amiga Workbench 1.3 è copyright 1985, 1986,1987,1988 Commodore-Amiga Inc., tutti i diritti sono riservati.

Per ogni questione tecnica riguardante il funzionamento del disco telefonare il Martedì e il Giovedì dalle ore 14.30 alle ore 16.00 al 02-6948287.

Come entrare in CLI

- Accendere il computer
- Inserire il disco con il Workbench
- Attendere il caricamento
- Cliccare due volte sul disco del Workbench
- Cliccare due volte sull'icona SHELL (per il nuovo CLI), oppure aprire il il cassetto System
- Selezionare l'icona CLI

Esempi di caricamento da CLI
- Entrare in CLI
- Digitare: COPY FROM C: TO RAM: PATH RAM::
- Inserire il disco di Amiga Magazine e digitare CD DF0:
- Digitare CD UTILITIES

A questo punto, per esempio, si può digitare:

- DSD

Amiga Magazine n. 15

SERVIZIO LETTORI **Compilare e spedire in busta chiusa a: GRUPPO EDITORIALE JACKSON Area Consumer - Via Pola, 9 - 20124 Milano**

A) Come giudichi questo numero di Amiga Magazine ?

❑ Ottimo
❑ Molto Buono
❑ Buono
❑ Discreto
❑ Sufficiente
❑ Insufficiente

B) Quale (i) articolo (i) o rubrica hai apprezzato di più?

______

Quale meno?

______

C) Cosa ti piacerebbe leggere nei prossimi numeri di Amiga Magazine ? ______

D) Ti è piaciuto il Software ?

______

E) Quante persone leggono la tua copia di Amiga Magazine ?

______

F) Possiedi un computer? Quale?

______

Quale (i) computer intendi acquistare in futuro?

______

G) Leggi altre riviste Jackson?

❑ SI ❑ NO

Quali? ______

H) Leggi altre riviste del settore?

❑ SI ❑ NO

Quali? ______

I) Oltre alle riviste dedicate al computer quali sono le tue letture preferite? ______

L) Quali sono i tuoi hobbies e maggiori interessi?

❑ Sport
❑ Musica
❑ Videoregistrazione
❑ Hi - Fi
❑ Fotografia
❑ Automobile
❑ Moto
❑ Viaggi

Nome ______
Cognome ______
Indirizzo ______
Età ______ **Professione** ______
Città ______
Prov. ______ **C.a.p.** ______ **Tel.** ______

GRUPPO EDITORIALE JACKSON
LIBRI DI TESTO PER
E PROFE
Il Gruppo Editoriale Jackson, leader da molti anni nella editoria tecnica specializzata, da sempre inserito nell'area della formazione professionale con il più completo e vasto catalogo libri esistente sul mercato italiano, dal 1987 sta lavorando ad un ambizioso progetto editoriale legato al mondo della Scuola Secondaria Superiore, che vuole garantire una produzione editoriale didatticamente avanzata, completa ed aggiornata in quelle discipline e in quegli indirizzi di studio più direttamente legati al mondo dell'Informatica, dell'Elettronica e delle Telecomunicazioni.
Thomas L. Floyd
CIRCUITI ELETTRICI
Corso di elettrotecnica generale
Cod. SD667-pp. 750-L. 39.500
Pietro Adorni
ELETTROTECNICA GENERALE
Cod. SD607-pp. 432-L. 26.000
Dino Pellizzaro
MISURE ELETTRICHE
Cod. SD565-pp. 440-L. 27.000
Stephen J. Chapman
MACCHINE ELETTRICHE
Cod.: SE885-pp. 610-L. 42.000
Mauro Gargantini
FONDAMENTI DI ELETTRONICA ANALOGICA
Dispositivi, circuiti e applicazioni
Cod. SE769-pp. 896-L. 41.000
Ramakant A. Gayakwad
AMPLIFICATORI OPERAZIONALI E CIRCUITI INTEGRATI LINEARI
Cod. SE897-pp. 690-L. 39.000
Mauro Gargantini
Armando Zecchi
ELETTRONICA INTEGRATA LINEARE
Cod. SD490-pp. 392-L. 25.000
Herbert Taub
Donald Schilling
FONDAMENTI DI ELETTRONICA INTEGRATA DIGITALE
Cod. SD493-pp. 308-L. 26.000
Ronald J. Tocci
SISTEMI DIGITALI
Corso di elettronica digitale
Cod. SE890-pp. 980-L. 43.000
Pierfranco Ravotto
Eugenio Piana
PROGETTARE CON L'ELETTRONICA DIGITALE
Dalla logica cablata al programmabile
Cod. SD620-pp. 640-L. 38.000
Giuseppe Giuliano
MICROPROCESSORI
II Edizione
Architettura e programmazione
Cod. SE622-pp. 330-L. 25.000
Edward A. Maruggi
DISEGNO ELETTRONICO
Tecniche grafiche per la progettazione
Cod. SE645-pp. 368-L. 27.000
Fosco Bellomo
ELEMENTI PASSIVI TECNOLOGIE E DISPOSITIVI
Corso di tecnologie elettroniche - Vol. I
Cod. SD613-pp. 356-L. 27.000
Fosco Bellomo
ELEMENTI ATTIVI: TECNOLOGIE E DISPOSITIVI
Corso di tecnologie elettroniche Vol. II
Cod. SE892-pp. 380-L. 28.000
Fosco Bellomo
MICROELETTRONICA NUOVE TECNOLOGIE
Corso di tecnologie elettroniche - Vol. III
Cod. SD651-pp. 224-L. 21.000
Renzo Traversini
MICROELETTRONICA: TECNOLOGIE E DISPOSITIVI
Cod. SD492-pp. 192-L. 20.000
Vincenzo Favale
ELEMENTI DI ELETTROTECNICA
Corso di elettronica Vol. I
Cod. SE895-pp. 354-L. 25.000
Vincenzo Favale
ELEMENTI DI ELETTRONICA LINEARE
Corso di elettronica Vol. II
Cod. SE755-pp. 350-L. 25.000
Vincenzo Favale
ELEMENTI DI ELETTRONICA DIGITALE E MICROPROCESSORI
Cod. SD754-pp. 476-L. 30.000
In vendita
le librerie sc

ADEGUATI AI PROGETTI AMBRA ED ERGON

GLI ISTITUTI TECNICI SSIONALI

Paul B. Zbar
Joseph G. Sloop
LABORATORIO DI ELETTROTECNICA
Cod. SD497·pp. 296·L. 25.000

Paul B. Zbar
Joseph G. Sloop
LABORATORIO DI ELETTRONICA DI BASE
Cod. SD496·pp. 272·L. 22.500

Paul B. Zbar
Joseph G. Sloop
LABORATORIO DI ELETTRONICA INTEGRATA
Cod. SD495·pp. 246·L. 22.000

Paul B. Zbar
Joseph G. Sloop
DALL'ELETTRONICA ALL'ELETTRONICA INTEGRATA
Manuale di laboratorio
Cod. SD664·pp. 760·L. 48.000

Franco Sgorbani
Luciana Donelli
SISTEMI, AUTOMAZIONE E LABORATORIO
Cod. SE747·pp. 508·L. 39.000

Mario Malcangi
SISTEMI, MODELLI E PROCESSI
Corso di sistemi d'automazione - Vol. I
Cod. SD598·pp. 196·L. 20.000

Mario Malcangi
SISTEMI DIGITALI PER L'AUTOMAZIONE
Corso di sistemi d'automazione - Vol. II
Cod. SD599·pp. 248·L. 20.000

Mario Malcangi
SISTEMI, AUTOMAZIONE E CONTROLLO
Corso di sistemi d'automazione - Vol. III
Cod. SD491·pp. 192·L. 20.000

Felice Tarantini
TELEFONIA
Telefonia di base, sistemi di trasmissione, commutazione telefonica
Cod. SE666·pp. 622·L. 39.000

Ugo Sgubbi
Alessandro Gava
Giuseppe Saccardi
TELEMATICA
Corso teorico e pratico di trasmissione dati
Cod. SE879·pp. 474·L. 34.000

Paul H. Young
COMUNICAZIONI ELETTRICHE
Corso di radioelettronica
Cod. SD633·pp. 500·L. 37.000

Massimo Tranchina
TELECOMUNICAZIONI
Dal segnale alla trasmissione dati
Cod. SE869·pp. 280·L. 25.000

Peter Bishop
INFORMATICA GENERALE VOL. I
Cod. SE893·pp. 272·L. 20.000

Peter Bishop
INFORMATICA GENERALE VOL. II
Cod. SE894·pp. 334·L. 24.000

Mariangela Botti
Roberto Ranzani
DAL PROBLEMA AL PROGRAMMA
Cod. SD494·pp. 328·L. 26.000

Pierre Le Beux
PASCAL E TURBO PASCAL
II Edizione
Cod. SD775·pp. 384·L. 30.000

Pierre Le Beux
BASIC
Cod.: SD774·pp. 408·L. 30.000

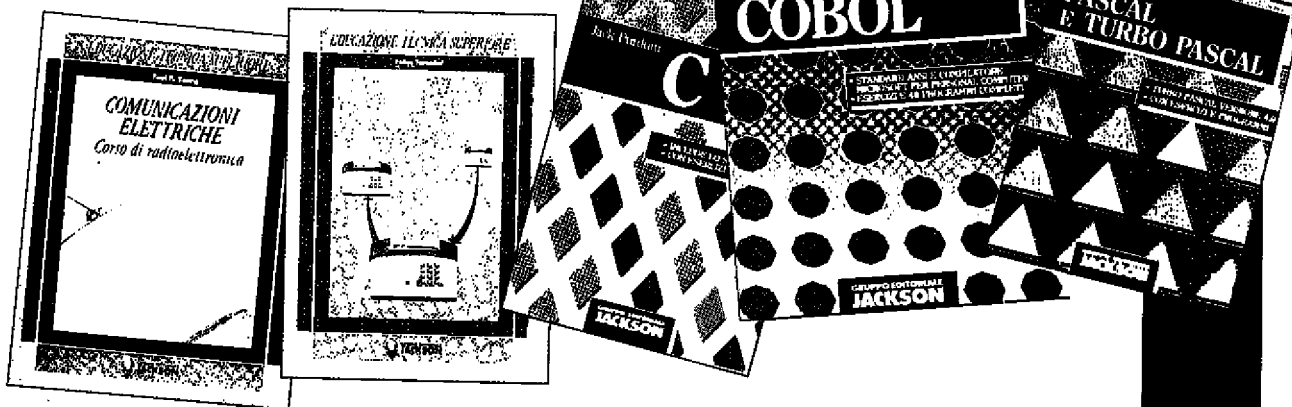
Kip R. Irvine
COBOL
Cod. SE899·pp. 564·L. 42.000

Jack Purdum
C
Cod. SE898·pp. 456·L. 35.000

Roberto Doretti
Roberto Farabone
FORTRAN
II Edizione
Cod. SD770·pp. 388·L. 34.000

Salvatore Consentino
ORGANIZZAZIONE INDUSTRIALE STUDI DI FABBRICAZIONE E DISEGNO
Cod. SD624·pp. 220·L. 24.000

Nuovi strumenti per una scuola che cambia

in tutte
olastiche

Dal 6502 al 68000

Wesley Howe

Se sapete programmare in linguaggio macchina, è probabile che abbiate imparato su un Commodore 64, un Atari a 8 bit, un Apple II o un altro microcomputer con 6502. Tuttavia, ora che possedete un Amiga forse vi piacerebbe tentare di mettere le mani nella programmazione del 68000.

Dal momento che il 68000 è molto più complesso del 6502, dovrete imparare molte cose prima di poter affrontare il complesso sistema operativo di Amiga. Per rendere le cose più facili, intendo descrivere il 68000 in termini che siano già familiari al programmatore che abbia esperienza con il 6502. Invece di esaminare concetti di base come bit, byte e registri, esamineremo alcune delle differenze fondamentali che esistono fra l'assembler del 68000 e quello del 6502. Ciò che non voglio tentare è una spiegazione analitica di tutte le istruzioni del 68000. C'è più di un libro dedicato a questo, uno dei miei preferiti è 680x0 Programming by Example di Stan Kelly-Bootle (Howard W. Sams & Co., 1988).

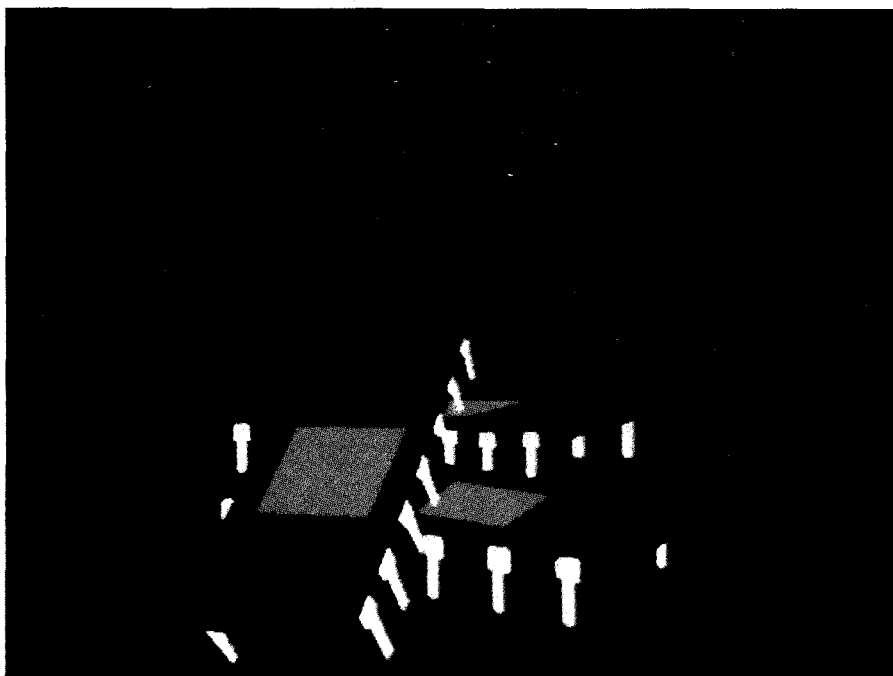
Più registri

La differenza più ovvia tra il 68000 e il 6502 sta nel numero di registri hardware. Il 68000 ha non pochi registri in più da gestire e sono anche organizzati in maniera diversa (si veda la tabella 1).

Al posto di un accumulatore, il 68000 ha otto registri dati chiamati D0-D7. E i registri indice del 6502, X e Y, sono rimpiazzati da sette registri indirizzo cui ci si riferisce come A0-A6 (c'è un ottavo registro indirizzi, A7, che serve come stack pointer). Nonostante il numero di registri disponibili sia superiore di tre volte, il set di istruzioni del 68000 è in realtà piuttosto semplice da gestire. Di fatto, una volta che vi sarete abituati ai registri aggiuntivi, sarà difficile programmare senza di essi.

Organizzazione della memoria

Il 6502 può indirizzare 64K di memoria mediante il suo program counter a 16 bit. La famiglia di microprocessori del 68000, con i loro program counter a 32 bit, è in grado di indirizzare quattro Gigabyte di memoria su alcuni modelli (68020 e 68030), ma il 68000 standard ha di fatto solo 24 linee indirizzi in uscita dal chip, cosa che limita a 16 Megabyte l'area di indirizzamento. Questa è comunque 256 volte più grande della capacità di memoria del 6502. Il 68000 vede questa memoria in maniera analoga al 6502, come un array di byte numerati da 0 fino al limite massimo ma, sul 68000, il limite massimo è più alto.

Il 68000 non stabilisce nessuna particolare area di memoria per lo stack o la pagina 0. Lo stack può essere ovunque in RAM e non è limitato a 256 byte; può avere qualsiasi lunghezza purché non entri in conflitto con altri programmi. I modi di indirizzamento della pagina 0, come LDA ($FB),Y, non sono necessari, perché questo tipo di indirizzamento può essere effettuato con registri indirizzo sul 68000. Sulla maggior parte delle implementazioni del 68000, tuttavia, i 1024 byte più bassi della memoria sono riservati per essere usati come vettori di

exception e di interrupt.
Una differenza realmente significativa nel 68000 è che degli elementi composti da più byte, come le word (valori di due byte) e le longword (valori di 4 byte), sono posti in memoria e recuperati prima i byte più alti, e poi quelli più bassi. Questo significa che porre in memoria una word con il valore di $1234 all'indirizzo $0440, si tradurrà in un $12 nella locazione $0440 e in un $34 nella locazione $0441. Questo è l'opposto del metodo standard byte-basso/byte-alto usato dal 6502 per porre numeri a 16 bit in memoria. Siccome il 68000 è un chip a 16 bit reali, tutte le sue istruzioni occupano 16 bit (due byte) o multipli di 16 bit. Queste istruzioni devono iniziare a un indirizzo di memoria pari (indirizzo $08F2, per esempio, e non $08F1 o $08F3). Il vostro assembler dovrebbe comunque impedirvi di infrangere questa regola. Inoltre, per potere accedere ai valori delle word e delle longword come a qualcosa di unitario, è necessario che esse siano poste ad un indirizzo pari. Altrimenti, saranno trattate come byte separati e senza relazione tra loro. Ahimè, è molto facile violare quest'ultima regola, se non si sta attenti.
Si veda la tabella 2, per un paragone fra i vari modi di indirizzamento dei due microprocessori.

Sintassi

Il linguaggio assembly del 68000 appare piuttosto diverso da quello del 6502. Le mnemoniche, gruppi di lettere usate per rappresentare le istruzioni del linguaggio macchina, possono essere più lunghe di tre caratteri e la maggior parte consente l'uso di un indicatore di formato che stabilisce se l'istruzione si riferisce al byte, alla word o alla longword del dato. Per specificare un indicatore di formato, inserite un punto immediatamente dopo l'istruzione e fatelo seguire da una lettera singola che specifichi il formato dei dati desiderato, B per il byte, W per la word e L per la longword. Analogamente, le istruzioni di salto possono apparire con una estensione .S per indicare salti corti (un salto che usa uno spiazzamento a 8 bit per saltare fino a 127 byte in avanti o fino a 128 byte all'indietro).
Molte istruzioni del 68000 richiedono due operandi, separati da una virgola. Il primo è detto operando sorgente; il secondo, operando destinazione. Altri microprocessori che usano due operandi, come l'8086, seguono l'ordine opposto. Sul 68000, tuttavia, l'ordine è sempre Sorgente, Destinazione. Per esempio, MOVEQ.L #0,D1 pone uno 0 nel primo registro dati. Parentesi e virgole giocano un ruolo significativo fra gli operandi del 68000, separando i diversi elementi e permettendo all'assembler di determinare il modo di indirizzamento adeguato.
Si vedano le tabelle 3 e 4 per un confronto fra le istruzioni del 6502 e quelle del 68000.

Assemblare e usare il linker

Differenze a livello di memoria, assieme alla capacità di Amiga di effettuare il multitasking, aggiungono a un assembler per Amiga cose con

Tabella 1: Registri

Un breve confronto fra i registri del 6502 e quelli del 68000. Il numero di registri per ciascun tipo appare in parentesi quadre ([]).

6502	68000
[1] 8 bit accumulatore (A)	[8] 8/16/32 bit registri dati (D0-D7)
[2] 8 bit registri indice (X e Y)	[7] 32 bit registri indirizzi (A0-A6)
[1] 16 bit program counter (PC)	[1] 32 bit program counter (PC)
[1] 8 bit stack pointer (S)	[2] 32 bit stack pointer (SP o A7)*
[1] 8 bit registro di stato (SR)	[1] 8/16 bit registro dei codici di condizione (CCR)

* Il registro indirizzi A7 è lo stack pointer. A7, in realtà, consiste di due registri, ma solo uno può essere attivo in un dato istante.

Tabella 2: Modi di indirizzamento

Questa tabella mette a confronto i modi di indirizzamento delle istruzioni del 6502 con quelli del 68000, usando i nomi standard di questi modi.

6502	68000
Immediato di memoria	Immediato
Diretto da memoria	Assoluto
Implicito	Implicito
Accumulatore	Registro dati diretto
Indiretto preindicizzato	Indiretto da registro indirizzi con spiazzamento e indice*
Indiretto postindicizzato	Indiretto da registro indirizzi con spiazzamento e indice*
Indirizzamento indicizzato	Indiretto da registro indirizzi con spiazzamento*
Indirizzamento indiretto	Indiretto da registro indirizzi*
Indirizzamento relativo	Relativo al Program Counter
(nessun equivalente)	Registro indirizzi diretto
(nessun equivalente)	Relativo al Program Counter con spiazzamento
(nessun equivalente)	Relativo al Program Counter con spiazzamento e indice
(nessun equivalente)	Indiretto da registro indirizzi con incremento successivo
(nessun equivalente)	Indiretto da registro indirizzi con decremento precedente

* Non è l'esatto equivalente

cui i programmatori del 6502 non hanno mai avuto a che fare. Al fine di effettuare il multitasking, per esempio, i programmi per Amiga devono poter essere caricati e lanciati a qualsiasi indirizzo. Sebbene il 68000 possieda modi di indirizzamento in grado di produrre codice totalmente rilocabile, non dovete limitarvi a questi modi per creare un programma adatto al multitasking. Amiga si prende cura di questo attraverso il suo loader.

L'assembler per Amiga crea file oggetto in cui i riferimenti agli indirizzi di memoria vengono lasciati vuoti. A questi file sono aggiunte delle informazioni che, dopo la fase di link, permettono al loader, che ha il compito di caricare i programmi in memoria, di porre i valori appropriati nei riferimenti lasciati vuoti, prima di lanciare il programma. Le informazioni per effettuare questo lavoro, contenute nei cosiddetti hunk_reloc (hunk di rilocazione), sono costituite da offset che indicano la posizione di memoria desiderata. Quando un programma viene caricato, gli indirizzi dei riferimenti mancanti vengono calcolati aggiungendo gli offset all'indirizzo in cui il codice è stato caricato. Così, quando il programma è pronto per essere eseguito, ogni istruzione in precedenza incompleta farà riferimento a una locazione assoluta di memoria.

Usando la direttiva assembler SECTION (sezione), potete spezzare il vostro programma per Amiga in unità separate. Segmentando il vostro codice, permetterete ad Amiga di effettuare lo scatter load (caricamento discontinuo) del programma sezione per sezione: non sarà più necessario caricarlo in un'area contigua di RAM. Ciò rende più facile per il computer lanciare un programma quando la memoria scarseggia. Sezioni differenti possono fare riferimento alle routine e alle locazioni di memoria l'una dell'altra, anche se si trovano in aree di memoria separate. Non siete obbligati a dividere il vostro programma in segmenti. Tuttavia, in ampi progetti di programmazione, questo è spesso l'unico modo pratico per

Tabella 3: Istruzioni

La tavola che segue fornisce, per ciascuna istruzione del 6502, l'equivalente del 68000. Siccome i due processori sono differenti quanto a capacità di indirizzamento, non vengono elencati i modi indicizzati e quelli relativi alla pagina del 6502. La lettera minuscola n rappresenta il numero di registro (la combinazione di lettere Dn, per *esempio, rappresenta tutti i registri D0-D7).*

6502	68000	Commenti sul 68000
ADC #$34	ADDX.B #$34,Dn	Differenze funzionali nei flag
AND $4046	AND.B $4046,Dn	
ASL A	ASL.B Dn	
BCC etichetta	BCC.S etichetta	Può anche effettuare salti lunghi
BCS etichetta	BCS.S etichetta	
BEQ etichetta	BEQ.S etichetta	
BIT $4049	BTST.B Dn,$4049	
BMI etichetta	BMI.S etichetta	
BNE etichetta	BNE.S etichetta	
BPL etichetta	BPL.S etichetta	
BRK	TRAP #0	Una delle 8 trap possibili
BVC etichetta	BVC.S etichetta	
BVS etichetta	BVS.S etichetta	
CLC	ANDI.B #$FE,CCR	Il flag di carry è il bit #0
CLD	(nessun equivalente)	Non c'è il flag decimale
CLI	ANDI.W #F8FF,SR	Un'istruzione privilegiata
CLV	ANDI.B #$FD,CCR	Il flag di overflow è il bit #1
CMP #$20	CMPI.B #$20,Dn	
CPX #0	CMPA.W #0,An	Non c'è equivalente per il byte (.B)
CPY $404E	CMPA.W $404E,An	E' richiesto un indirizzo pari
DEC $4049	SUBQ.B #1,$4049	Può sottrarre valori compresi tra 1 e 8
DEX	SUBQ.L #1,An	Può sottrarre valori compresi tra 1 e 8
DEY	SUBQ.L #1,An	Può sottrarre valori compresi tra 1 e 8
EOR #$FF	EOR.B #$FF,Dn	
INC $4049	ADDQ.B #1,$4049	Può sommare valori compresi tra 1 e 8
INX	ADDQ.L #1,An	Può sommare valori compresi tra 1 e 8
INY	ADDQ.L #1,An	Può sommare valori compresi tra 1 e 8
JMP $4600	JMP $4600	Deve saltare a un indirizzo pari
JSR $4800	JSR $4800	Deve saltare a un indirizzo pari
LDA #'A'	MOVEQ #'A',Dn	Influenza tutto il registro a 32 bit
LDX $24	MOVE.W $24,An	Non c'è equivalente per il formato byte
LDY $4080	MOVE.W $4080,An	Non c'è equivalente per il formato byte
LSR A	LSR.B Dn	
NOP	NOP	
ORA #4	ORI.B #4,Dn	
PHA	MOVE.B Dn,-(SP)	
PHP	MOVE.W CCR,-(SP)	
PLA	MOVE.B (SP)+,Dn	
PLP	MOVE.W (SP)+,CCR	
ROL A	ROL.B #1,Dn	Puo ruotare da 1 a 8 volte
ROR A	ROR.B #1,Dn	Puo ruotare da 1 a 8 volte
RTI	RTE	
RTS	RTS	
SBC #1	SUBX.B #1,Dn	
SEC	ORI.B #1,CCR	
SED	(nessun equivalente)	Non c'è il flag decimale
SEI	ORI.W #$0700,SR	Un'istruzione privilegiata
STA $2034	MOVE.B Dn,$2034	
STX $2034	MOVE.W An,$2034	Non c'è equivalente per il formato byte
STY $20	MOVE.W An,$20	Non c'è equivalente per il formato byte
TAX	MOVEA.W Dn,An	Non influisce sui flag
TAY	MOVEA.W Dn,An	Non influisce sui flag
TSX	MOVEA.L SP,An	Non influisce sui flag
TXA	MOVE.W An,Dn	Non c'è equivalente per il formato byte
TXS	MOVEA.L An,SP	Non influisce sui flag
TYA	MOVE.W An,Dn	

costruire un programma.
Per rendere visibile ad altri una porzione del segmento, dovete usare la direttiva XDEF (eXternal DEFinition, definizione esterna) seguita dal nome dell'etichetta. L'assembler tratta queste etichette aggiungendo il loro nome e il loro offset (distanza in byte dall'inizio della sezione) alla fine della sezione, nel proprio file di output (che viene detto hunk quando si trova unito correttamente ad altre sezioni del programma). Quando avete bisogno di chiamare una routine o di accedere a dei dati localizzati in un altro modulo, usate una direttiva XREF (eXternal REFerence, riferimento esterno) in un punto che preceda l'uso dell'etichetta (normalmente vengono raggruppate all'inizio del file sorgente). Infine, potrete usare queste etichette come operandi nel vostro codice.
Il linker, un programma sconosciuto alla maggior parte degli utenti del 6502, esamina i file oggetto creati dall'assembler e confronta tutte le definizioni esterne con i corrispondenti riferimenti esterni (da qualche parte, ci deve essere una definizione per ogni riferimento; se il linker non riesce a trovarne una, riceverete un messaggio di errore). Tutti i riferimenti sono controllati e convertiti in riferimenti di rilocazione degli hunk per il loader. Il linker termina il lavoro copiando il codice e i dati contenuti in tutti i file oggetto (possono essere più d'uno) in un unico file di output, aggiungendovi le rilocazioni degli hunk appena create e quelle già esistenti e infine eliminando le informazioni sui riferimenti esterni e, possibilmente, sulle definizioni esterne (queste possono anche essere conservate, in modo che i debugger riescano ad accedere alle informazioni sui simboli da un programma eseguibile).
Siccome possiamo costruire il nostro programma per pezzi, è possibile creare una collezione di routine già assemblate da passare al linker. Ciò vi farà risparmiare il tempo necessario per riassemblare la routine ogni volta che la usate. Quando queste collezioni di routine preassemblate, dette librerie linked, vengono passate al linker, questo le esamina ed estrae solo le routine cui avete fatto riferimento nel vostro programma. Amiga è in grado di indirizzare dati e programmi in maniere distinte. Sebbene su Amiga questi due spazi di indirizzamento, potenzialmente separati, siano combinati, i concetti sezione dati e sezione di codice sono stati trasferiti negli strumenti e nella pratica di programmazione di Amiga. Infatti, ci sono tre tipi di hunk che indicano il tipo di memoria: CODE (qualche volta detto anche TEXT), DATA e BSS ("Bulk Storage Space", quantità di spazio riservato). Le sezioni DATA sono usate per mantenervi locazioni di memoria inizializzate, mentre BSS è semplicemente una sezione di memoria riservata perché la usi il vostro programma. A differenza di quanto accade con la memoria DATA, non c'è alcuna garanzia sul contenuto della memoria BSS quando il programma viene lanciato. Tenete presente che la memoria BSS non fa parte del programma su disco, è allocata quando il programma viene lanciato.

Tentate

Il modo migliore per imparare a programmare, consiste nella pratica effettiva. Sebbene non abbia spiegato l'effettiva costruzione di un programma per il 68000, penso che i molti articoli e libri che si occupano del linguaggio assembly per Amiga vi appaiano ora un poco più comprensibili. ▲

Attualmente Wesley Howe è il direttore dell'ingegneristica presso la compagnia telefonica della North Carolina; di notte, è un avido programmatore del 68000. Wesley è l'autore del popolare assemblatore per Amiga Cape 68k e di Inovatools II, entrambi disponibili presso la Inovatronics.

Tabella 4: dal 68000 al 6502

Questa tabella mostra come alcune delle più comuni istruzioni del 68000 vengano tradotte sul 6502. Come potete vedere, ci vogliono normalmente molte istruzioni del 6502 per svolgere il compito di una sola istruzione del 68000. Questa tabella non costituisce in nessun senso una lista completa: molte istruzioni del 68000 non possono affatto essere riprodotte dal 6502.

68000	6502
ABCD.B D0,D1	SED
	CLC
	ADC $1234
ABCD.B -(A0),-(A1)	SED
	CLC
	DEX
	LDA ($1234,X)
	ADC ($2345,X)
	STA ($2345,X)
BCHG.B #2,$1234	LDA $1234
	EOR #$04
	STA $1234
BCLR.B #0,2345	LDA $2345
	AND #$FE
	STA $2345
BSET.B #7,$4567	LDA $4567
	ORA #$80
	STA $4567
CLR.L $1234	LDA #0
	STA $1234
	STA $1235
	STA $1236
	STA $1237
DBRA D0,etichetta	SEC
	LDA $1235
	SBC #1
	STA $1235
	LDA $1234
	SBC #0
	STA $1234
	BCS etichetta
MOVE.W $1234,$12AE	LDA $1234
	STA $12AE
	LDA $1235
	STA $12AF
NEG.B $2345	SEC
	LDA #0
	SBC $1234
	STA $1234
NOT.B $1234	LDA $1234
	EOR #$FF
	STA $1234
PEA $0400	LDA #0
	PHA
	LDA #4
	PHA
SEQ $2400	BEQ L1
	LDA #FF
	STA $2400
	BNE L2
L1	LDA #0
	STA $2400
L2	...

Amiga '90 una fiera a Basilea

Tantissime novità dalla simpatica fiera svizzera

Carlo Cattoni

Si è svolta a Basilea, dal 9 al 12 Maggio, Amiga '90, una manifestazione dedicata interamente ad Amiga, sulla falsariga delle altre fiere che si sono tenute in Europa e negli Stati Uniti.
La cittadina svizzera è molto facile da raggiungere sia dall'Italia, che dalla Francia e Germania, e si è rivelata molto accogliente sotto tutti i punti di vista.
Appena arrivati, ci siamo immersi nella folla di visitatori alla ricerca di novità interessanti.
Al centro del padiglione troneggiava lo Stand Commodore, nel quale, oltre a gadget quali magliette (I Love Amiga...), borse e ombrellini, era presente un ormai mitico Amiga 3000 con applicazioni Multimedia, circondato perennemente da un capannello di persone entusiasmate dalle prestazioni della nuova macchina. Consistente era la cospicua presenza di rivenditori di hardware, con stand stracolmi di espansioni di memoria da 0 a 8 Mb, hard disk su scheda e mille gadget per Amiga; tappetini per mouse fluorescenti e mouse da autocolorare facevano, infatti, un po' di colore tra le schede. Tra le novità, un incredibile hard disk da 20 Mb interno per A500, con controller SCSI, da 2 pollici e mezzo. Interno, direte voi? Ebbene sì, è grande come un mouse, più sottile, autoboot, tempo di accesso 23 ms, 400 Kb di transfer rate e il prezzo a cui verrà pubblicizzato si profila molto interessante.
Unica difficoltà, l'installazione, da farsi ovviamente presso un rivenditore competente.
Le schede acceleratrici non mancavano, a partire da una Hurricane H2000 con 68020 a 14 MHz fino ad una 68030 a 28 MHz per Amiga 2000, e una nuovissima Hurricane H500 con 68020 a 14 MHz per il piccolo 500; presenti anche i rappresentanti tedeschi della GVP (che ha portato una 68030 a 33 MHz e ha annunciato una scheda a 50 MHz).
Gli svizzeri hanno pensato a coloro chi ha bisogno della compatibilità IBM e che deve usare le Janus con la lentezza che le contraddistingue, e così hanno "sfornato" due schede velocizzatrici, una per la XT a 8 MHz e una per la AT a 12 MHz.
Altre "grandi" società americane presenti alla manifestazione erano

Lo stand Commodore

Gli stand erano stracolmi di software e hardware

Il fantastico stand della Gigatron

la Vidtech che presentava lo Scanlock VSL 1P, genlock Super VHS dalle ottime caratteristiche e dal prezzo molto interessante; la IVS con i controller SCSI Trumpcard, le espansioni Meta 4 a moduli SIMM e le doppie interfacce parallele "intelligenti" Printerface.
Inoltre, la IVS presentava la Trumpcard Professional, che promette di essere il controller SCSI con il più elevato transfer rate mai riscontrato su un Amiga (1.5 Mb/sec su una macchina con il 68030, 900 Kb/sec su una macchina standard).
L'applicazione demo per la Trump Professional era impressionante: uno spezzone lungo oltre 600 fotogrammi di un cartone animato di Walt Disney, era caricato da un Hard Disk Quantum, con tempo di accesso di 11 msec., e riprodotto in tempo reale!
Tutti i prodotti IVS, che vantano un ottimo rapporto prezzo/prestazioni, sono distribuiti dalla Digimail di Milano. La Supra Corporation presentava, presso lo stand del proprio distributore tedesco, i noti controller per HD, i modem interni e le espansioni per tutti gli Amiga.
In campo video abbiamo ammirato il LIVE!2000 con il software Invision per gli effetti speciali realtime di screen splitting, solarizzazione, pixelizzazione e zoom, ottimo per la postproduzione video.
Unico espositore italiano alla mostra era la Computer Center con le note schede di espansione di memoria per Amiga 500 (esterne e interne) e 2000, e i controller HD economici.
Praticamente tutti gli stand dei rivenditori avevano un nutrito reparto di software originale, dove era possibile acquistare le ultime versioni dei pacchetti professionali e dei game più famosi; evidentemente fuori dall'Italia il mercato è molto meno inquinato dai rivenditori del "dischetto a 5000 lire", vera piaga del panorama commerciale nostrano.
Anche il software di pubblico dominio aveva il suo spazio, in particolare alla COMP-U-SAVE che presentava un catalogo di oltre 1200 dischi, con la collezione completa di Fred Fish e altre (FAUG, AMICUS...) confermando la validità dell'iniziativa di Amiga Magazine sul Public Domain. In campo Desk Top Publishing le configurazioni complete di scanner e stampanti facevano bella mostra e tenevano il passo con il Macintosh. Dobbiamo confessare di non essere tornati a casa a mani vuote, a tanta ricchezza d'hardware e software non si può certo restare indifferenti.
Speriamo che anche in Italia, al prossimo Amiga Days, i nostri rivenditori riescano a presentare novità e materiale interessante quanto quello che abbiamo visto a Basilea ! ▲

Pen Pal

Un super word processor con un potente database

Rhett Anderson

Pen Pal è un word processor con molte sorprese. La sorpresa più significativa è la qualità dell'output che Pen Pal può generare su una buona stampante a colori. Pen Pal è un programma di "sfondamento" per Amiga, perché vi permette di miscelare i font Near-Letter-Quality (NLQ) della vostra stampante con la grafica di Amiga. I precedenti word processor a colori, come Pro Write, vi facevano scegliere tra i font NLQ della vostra stampante e i font e la grafica di Amiga. Tuttavia, la versione 2.5 di Pro Write aggiunge questa nuova funzione, oltre ad altri cambiamenti minori.
Un'altra gradita sorpresa, è l'inclusione di un database. Non è Superbase, ma è utile: ideale, di fatto, per la maggior parte degli usi di un database. Anche se mi piacciono i database, mi domando se sia una buona idea quella di combinare un word processor e un database. Non voglio dover caricare tutto il codice extra del database, ogni volta che intendo scrivere una lettera.
Pen Pal sembra essere "ghiotto" di memoria. Ho dovuto eliminare la maggior parte dei cache buffer del mio hard disk per poterlo lanciare. Quando la memoria è scarsa non risponde bene. Anche se il programma richiede un Megabyte di RAM, opera molto meglio con il doppio. Anche la lunghezza dei file è eccessiva. Di fatto la veloce ispezione di un file di testo mostra che a ciascun carattere corrisponde un altro byte, forse un byte di attributi. Questo formato sembra specifico a Pen Pal. Avrei apprezzato la capacità di caricare i file IFF TEXT salvati con Pro Write.
Il word processor in sé è molto funzionale. La sua velocità è ragionevole per un word processor WYSIWYG. I normali comandi di editing funzionano bene: la semplice pressione del pulsante del mouse muove il cursore, una doppia pressione seleziona una parola, una pressione tripla seleziona una linea. Le capacità grafiche di Pen Pal sono impressionanti. Potete importare file IFF, disegnare linee colorate e riquadri, spostare o tagliare immagini. Il testo può girare facilmente attorno alle figure, ma non potete dividere il testo in più colonne. E un'omissione critica, perché impedisce la realizzazione di notiziari, qualcosa cui il word processor sembra, per altri aspetti, idealmente destinato.
Prima di dare il mio "benestare" a questo programma, lasciatemi elencare alcune osservazioni emerse durante l'uso del programma.
Quando ho lanciato per la prima volta Pen Pal, rimasi deluso dal fatto che lo schermo era costituito da caratteri neri su sfondo bianco. Sebbene qualcuno, con un monitor monocromatico o multisync, possa trovare quella soluzione piacevole, è sufficiente a far venire il mal di testa a coloro che usano un monitor Commodore. Se volessi uno sfondo bianco, lo prevederei mediante Preferences. La mia delusione si intensificò quando mi accorsi che ogni set alternativo di colori usava il bianco come colore di sfondo. Ho invertito manualmente il colore del testo con il colore di sfondo, per accorgermi che il bordo era bianco. Ancora peggio, non ho potuto salvare le mie preferenze riguardanti i colori: devo cambiarle ogni volta che lancio il programma. Se avessi il sorgente di Pen Pal, questo sareb-

be il primo problema che risolverei. L'interfaccia utente è in qualche caso mal progettata. Il selettore dei file è ingombrante. L'interfaccia dello spelling checker, naturalmente in inglese, è anche peggio. Non esiste alcuna funzione per ignorare un comando o saltare una sequenza di operazioni, così non saprei che cosa fare se volessi lasciare una parola così com'è senza aggiungerla al dizionario. Ho tentato "cancel", ma quell'opzione mi ha portato completamente fuori dallo spelling checker. Sono rimasto deluso dalla mancanza di tasti Amiga equivalenti ai comandi del menu. Ce ne sono alcuni, ma non quelli che desidero.

Non appena muovi il puntatore del mouse sui gadget che rappresentano gli strumenti, viene visualizzato del testo sotto il puntatore. Questo testo informa sul significato di ogni gadget. E' una funzione utile, ma un po' sconcertante. Gli utenti con una certa esperienza cercheranno un modo per eliminare queste informazioni dinamiche. Non l'ho ancora trovato.

Mi piace molto lo strumento per visionare la pagina. Esso mostra la pagina corrente proprio nel modo in cui verrà stampata (meno una notevole quantità di risoluzione, naturalmente) Potete vedere ancora il vostro testo nella finestra adiacente, ma, sfortunatamente, non potete editarlo finché non riselezionate lo strumento, eliminando la visualizzazione della pagina intera.

Mi piace il riquadro di stato. Vi dice se il vostro documento troverà posto sul disco che avete selezionato. Perché non lo fanno tutti i word processor? Sfortunatamente, il programma è profondamente convinto che nel RAM disk non ci sia mai un byte di memoria libera. Come risultato, Pen Pal non tenterà mai di salvare un file in RAM:. Tuttavia, il device RAM: è un po' bizzarro, dal momento che si contrae e si espande per adattarsi ai vostri dati, ma non avevo mai visto prima un programma che non possa salvarvi i propri file.

Vi sono alcune stranezze che esiterei a definire come veri e propri bug. Per esempio, sono riuscito a selezionare un gadget che non era un gadget toggle. Sembrò corrompersi, ma non causò un Guru.

La flessibilità di Pen Pal è impressionante. Permette di modificare così tante variabili da far girare la testa. Per esempio, potete visualizzare le date in una quantità di formati che mai avevo visto. Potete stabilire la velocità del mouse. Potete scegliere ogni immaginabile opzione di formato. In alcuni casi, Pen Pal fa le cose così bene che mi piacerebbe vedere certe innovazioni in altri programmi. Per esempio, i font sono trattati in maniera perfetta. Un menu elenca i font per ampiezza, un altro per nome. Se avete più font di quanti possono stare in un menu, un altro menu di font apparirà automaticamente. Molto bello!

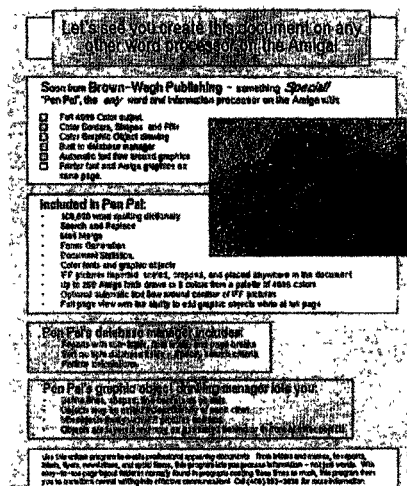
Il manuale è molto, molto buono. Quasi ogni problema che ho incontrato, cominciando ad usare il programma, ha ricevuto una risposta non appena ho aperto il manuale. Avrei desiderato che il manuale avesse una piccola sezione all'inizio che fornisse uno sguardo generale sul programma e le sue potenzialità. L'help in linea è stato meno utile, non mi è mai sembrato che fornisse la risposta che andavo cercando. Comunque, è bello vedere utilizzare il tasto Help.

Se avete bisogno di un database e di un word processor integrati, Pen Pal è una buona scelta. Potrete trasferire con facilità informazioni dal database a un documento. Pen Pal possiede anche un form manager che vi permetterà di creare documenti che possono essere modificati solo in campi predefiniti.

Pen Pal è all'altezza degli altri word processor per Amiga? E' certamente ambizioso quanto basta Sebbene abbia passato molti brutti momenti, quando ho usato la prima release del prodotto, la versione corrente (1.0m) sembra quasi stabile. Tuttavia, non scriverei un libro con Pen Pal. Userei Pen Pal per disegnare cartoline natalizie. Se fossi abbastanza organizzato per usare un database, potrei tenere sul database una lista di indirizzi.

Pen Pal si evolverà di sicuro, tuttavia. Dovrebbe dare ai concorrenti, nella già affollata arena dei word processor per Amiga, un nuovo motivo di preoccupazione. ▲

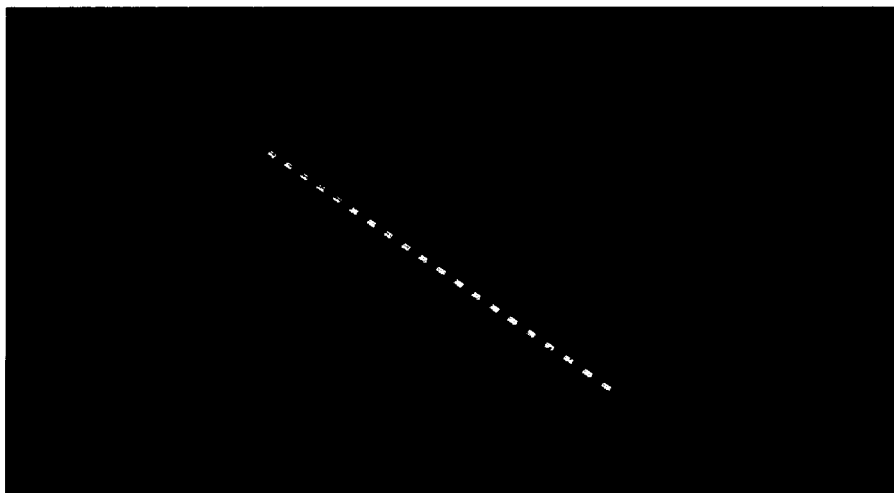
Ecco come appare una directory in Pen Pal

Impaginare con Pen Pal è molto semplice

Viaggio esplorativo alle... 8520

Tra tutti gli integrati utilizzati negli Amiga sicuramente gli 8520 sono i più conosciuti e menzionati.

a cura di Computer Lab
Via Cadore, 6 - Milano

Siamo ormai giunti al nostro quarto incontro e forse avremmo fatto un torto ai lettori di Amiga Magazine se, per diversi motivi, avessimo tralasciato di trattare l'argomento 8520.
Tale convinzione è supportata anche dal crescente numero di utilizzatori Amiga che, portando il proprio apparecchio presso il nostro laboratorio, al momento del ritiro, leggendo la scheda tecnica dell'intervento, esprimono assenso nel constatare che il motivo del guasto erano gli 8520.
Quanti di loro siano poi in grado di comprenderne il perché non sta certamente a noi scoprirlo, ma risulta evidente la volontà di conoscere l'interno del proprio computer nella speranza, di tutti, di non vederlo nuovamente in "difficoltà" di funzionamento. Inanzitutto iniziamo con il chiarire il significato della sigla CIA nata dalle iniziali di Complex Interface Adapter che tradotto può suonare come interfaccia complessa. L'integrato, o meglio, gli integrati 8520 svolgono per Amiga l'arduo compito di interfacciare il computer con l'esterno, provvedono cioè al controllo della tastiera, dei mouse, dei joystick permettono lo scambio dei dati con modem (porta seriale), stampanti (porta parallela), drive esterno e interno.
Per meglio gestire tutte queste funzioni i progettisti di Amiga, certamente non privi di fantasia nell'assegnare nomi ai diversi componenti, hanno pensato di utilizzarne due definendoli EVEN e ODD (pari e dispari - quasi a voler significare una sorta di spareggio per compiere le funzioni più "dure"!).
Un'altra importante funzione svolta dai due inseparabili 8520, è fornire al sistema un contatore che, ad esempio, assicura il corretto funzionamento del clock interno visualizzabile con la funzione contenuta nel Workbench. In figura 1 possiamo vedere i segnali gestiti dalle CIA sui 40 pin:

VSS - pin a massa o ground, 0 volt.
P.. - pin dal 2 al 17, rappresentano le linee di I/O (input/output) suddivise in due linee di 8 bit ciascuno. Le 8 linee, definite con PA da 0 a 7, provvedono ad inviare o ricevere i dati in correlazione delle altre 8 linee della porta B siglate PB.
PC - definisce l'uso delle due porte A e B appena viste: informa cioè se sono input od output e se lavorano congiuntamente a 16 bit.
TOD - è il segnale del clock.
VCC - è il pin di alimentazione dell'integrato: + 5 volt.
IRQ - Interrupt Request Output, è una linea che provvede a segnalare l'ordine di intervento dei diversi dispositivi.

R/W - è il segnale che, proveniente dalla CPU 68000, informa la CIA se deve inviare o ricevere segnali. Ciò avviene con l'invio di due diversi segnali, alto o basso corrispondenti a 0 o 5 volt.
CS - Chip Select, segnale presente su quasi tutti gli integrati, provvede ad abilitare, o meno, il funzionamento dell'integrato stesso.
FLAG - è la linea usata per sincronizzare le funzioni tra le 8520.
02 - è l'ingresso del clock utilizzato da tutti gli integrati TTL.
DB. - sono le 8 linee del DATA BUS, il "canale" di comunicazione tra i diversi integrati del computer. Su questa linea avviene il dialogo tra l'8520 e il resto di Amiga.
RES - linea di Reset dei registri della CIA.
RS.- sono le linee utilizzate per "programmare" il funzionamento dell'integrato.
SP e CNT - vengono utilizzate come timer per il funzionamento delle porte A e B.

Ora che abbiamo visto, a grandi linee, le caratteristiche dell'integrato, è utile vedere dove e come sono connesse le diverse porte utilizzate dal mouse, joystick, stampante eccetera..

I segnali pin per pin degli 8520

Iniziamo prendendo in esame la CIA definita ODD, siglata U7 per gli A500 e U300 per i 2000: alla porta B, quindi sui pin contrassegnati PB0-PB7, sono connesse le linee dati ad 8 bit della porta parallela, comunemente usata per stampanti.
Sulla porta A, pin da 2 a 7 indicati con PA0-PA7, sono indirizzati diversi segnali: dalle porte mouse il FIRE 0 e 1, dai drive le linee relative alla rilevazione della protezione disco e trk 0, mentre, per quanto concerne la tastiera, a questa CIA vengono inviati il clock, i dati e la linea del led power.
Riguardo al led, ricordiamo, per i meno esperti, che non si tratta di un comunissimo segnale per indicare se l'apparecchio è acceso o spento, ma, in caso di malfunzionamento del sistema, la frequenza di lampeggio può indicare, empiricamente, al tecnico quale sezione è in errore.
Ritornando alla porta parallela (Centronics per i tecnici) è utile considerare il funzionamento della linea PC, pin 18 della 8520, per

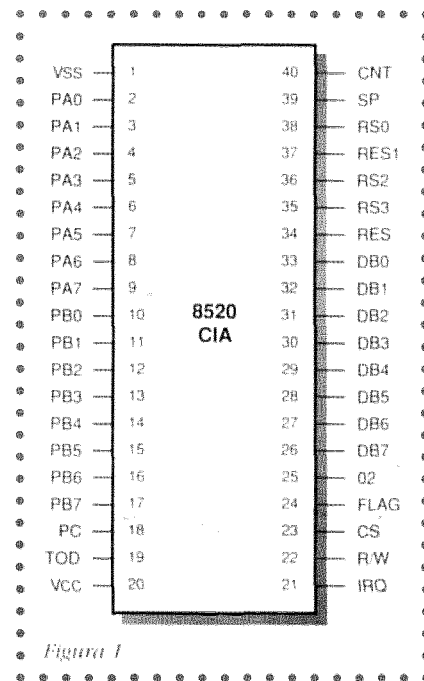



Figura 1

capire il funzionamento della acquisizione o invio dei dati.
A questo pin è inviato il segnale STROBE proveniente dalla stampante che, con i due livelli 0 e 1, indica alla 8520 se abilitare la porta A (PA0-PA7) a trasmettere dati, permettendo, quindi, l'inizio della fase di stampa.
Passiamo all'altra CIA, l'8520 EVEN per intenderci, siglata rispettivamente U8 e U301 negli Amiga 500 e 2000: alla porta A, pin da 2 a 9, vengono affidati i segnali di controllo della stampante parallela (ACK, POUT, BUSY, SEL) e tutte le linee relative alla porta seriale utilizzata per modem o altre connessioni con standard seriale RS232, con esclusione delle sole linee di trasmissione e ricevimento dati, conosciute come TXD e RXD gestite da PAULA.
In porta B, PB0 - PB7, la CIA gestisce i segnali di selezione dei rispettivi drive, massimo 4, e i controlli di avanzamento testine, rotazione motore e controllo DIR.
Nel precedente elenco di segnali interfacciati dalle rispettive CIA, abbiamo tralasciato, per motivi di chiarezza, di indicare come parte degli stessi segnali vengono reindirizzati su altri pin, per citarne alcuni il 18,
(segue a pag.77)

Un videogioco tutto nostro: trucchi e segreti (seconda parte)

In questa seconda parte, vedremo una serie di "trucchetti" che si dimostreranno molto validi per migliorare lo svolgimento delle azioni di gioco.

Gianni Biagini

Nello scorso numero, avrete certamente notato che gli spostamenti della pallina erano molto lenti, anche nella versione compilata.
Nasce spontaneo, quindi, chiedersi se esiste un metodo di sviluppo degli algoritmi di programmazione che sia conveniente sotto il profilo della velocità di esecuzione.
Premetto che forse i cultori della più pura programmazione strutturata "storceranno" il naso, purtroppo quest'ultima, essendo sicuramente più leggibile, risulta talvolta scomoda. Premettiamo che una buona conoscenza della macchina aiuta notevolmente nello sviluppo di procedure veloci; considerando, per esempio, che le condizioni multiple richiedono dei grossi tempi di elaborazione, possiamo trovare qualche sistema per evitarle o, almeno, renderle più familiari alla macchina.
Sicuramente qualcuno di voi ha già pensato di svolgere le funzioni logiche complesse, attraverso lo sviluppo in mappe di Carnaugh (nient'altro che delle comodissime tabelle che permettono di semplificare al massimo e di esprimere in termini di mintermini e di maxtermini qualsiasi funzione logica complesssa).
Purtroppo, spesso, questo è inutile, poiché raramente le espressioni logiche che incontriamo sono molto complesse, e generalmente i tempi di esecuzione della funzione sviluppata sono più lunghi.
Se noi riuscissimo in qualche modo a giostrare con le selezioni, vi assicuro che avremmo dei vantaggi rilevanti. Supponiamo di dover controllare un processo industriale dove i tempi di campionamento del dato siano importanti e che la macchina che controllo faccia uno sbuffo al secondo e un fischio ogni ora.
Supponiamo, inoltre, che la nostra procedura, se fischio e sbuffo coincidono, azioni il meccanismo A oppure il meccanismo B. Una comoda struttura di controllo potrebbe essere la seguente:

```
SE C'E' IL FISCHIO E SE C'E' LO SBUFFO
  ALLORA
     ATTIVA IL MECCANISMO A
  ALTRIMENTI
     ATTIVA IL MECCANISMO B
FINE SE
```

Benissimo, tutto funziona, però vi sono alcuni punti da analizzare.
Sappiamo che la macchina fischia ad intervalli di un'ora, minuto più minuto meno; sappiamo, inoltre, che i tempi di attivazione dei meccanismi A e B devono essere i più brevi possibili: risulta chiaro che la struttura appena esposta è fortemente penalizzante per i tempi di reazione.
Perché?
Semplicissimo, la macchina deve ogni secondo (questo è l'intervallo degli sbuffi) controllare 3 condizioni:

1) se la macchina sta fischiando
2) se la macchina sta sbuffando
3) se le condizioni sono entrambe vere.

Ragioniamo un attimo: il meccanismo B deve essere attivato circa una volta al secondo, mentre quello A circa una volta ogni ora; risulta evidente che il meccanismo B viene messo in funzione circa 3600 volte (60 secondi per 60 minuti) più del meccanismo A.

E se scrivessimo allora la struttura di prima utilizzando l'importante considerazione appena vista? Se invece di fare tre controlli ne facessimo uno solo ?
Questo è possibile:

```
SE LA MACCHINA SBUFFA
  ALLORA
    SE LA MACCHINA FISCHIA
      ALLORA
        ATTIVA IL MECCANISMO A
    FINE SE
  ALTRIMENTI
      ATTIVA IL MECCANISMO B
FINE SE
```

La differenza dalla vecchia procedura? Solo se la macchina sbuffa si esegue il secondo controllo (quello sul fischio).
Molti di voi staranno pensando che il guadagno è nullo poiché la macchina sbuffa una volta al secondo; avete ragione, ma la selezione che abbiamo appena implementato serve a far vedere come dei buoni criteri di progettazione riescano a potenziare di molto il codice.
Bastava scrivere:

```
SE LA MACCHINA FISCHIA
  ALLORA
    SE LA MACCHINA SBUFFA
      ALLORA
        ATTIVA IL MECCANISMO A
    FINE SE
  ALTRIMENTI
      ATTIVA IL MECCANISMO B
FINE SE
```

Capita la differenza? Solo se la macchina fischia (e quindi una volta ogni 3600 controlli circa) le selezioni che farò saranno due, altrimenti sarà solo una; quindi ho un risparmio di tempi molto elevato.
Il fatto che si sia parlato di macchine e,quindi, di processi industriali non è un caso, poiché l'informatica si avvicina sempre di più al mondo della meccanica e, in particolar modo, a quello della robotica; quella che vedremo sarà una buona tecnica di programmazione per i videogiochi e potrà essere applicata anche ad un processo industriale.
Ripensiamo all'esempio di prima alla luce delle nuove considerazioni: se invece di sbuffi e fischi abbiamo a disposizione missili che devono colpire astronavi nemiche e supponendo di non avere un rivelatore di collisione di sprite, dobbiamo controllare che le coordinate di missile e nemico coincidano; se la nostra astronave si trova sul fondo dello schermo (quello che gli americani chiamano "Bottom") e, viceversa, se il nemico si trova nella parte è necessario controllare che coordinate orizzontali e verticali coincidano ciò ci riporta alle tre selezioni di cui abbiamo precedentemente parlato. Una struttura del genere:

```
SE LE COORDINATE ORIZZONTALI E
   QUELLE VERTICALI COINCIDONO
   RISPETTIVAMENTE
  ALLORA
    SCRIVI "COLPITO"
  ALTRIMENTI
    SE LE COORDINATE VERTICALI
    COINCIDONO
      ALLORA
        SCRIVI "MANCATO"
    FINE SE
FINE SE
```

sebbene sintatticamente corretta, rappresenta gli stessi svantaggi di prima.
Molto meglio allora scrivere, ricordando che è più improbabile che il missile abbia già raggiunto la cima dello schermo, la seguente nuova versione:

```
SE LE COORDINATE VERTICALI
   COINCIDONO
  ALLORA
   SE LE COORDINATE ORIZZZONTALI
   COINCIDONO
     ALLORA
       SCRIVI "COLPITO"
     ALTRIMENTI
       SCRIVI "MANCATO"
   FINE SE
FINE SE
```

Siamo a buon punto, se le coordinate verticali non coincidono la procedura si sposta automaticamente a fine-se, processando altre linee di comando invece di perdere tempo in inutili selezioni.
Si consideri, inoltre, che in un videogioco, quando si spara ad un nemico, generalmente le coordinate orizzontali coincidono, quindi le tre selezioni si presentano spesso e volentieri.
Analizziamo un'altra struttura di selezione particolarmente vantaggiosa nel caso di scelte multiple che è presente nella maggior parte dei linguaggi di alto livello strutturati, il suo nome è CASE.
Premettiamo che le sintassi dei diversi linguaggi non sono sempre uguali, la stessa struttura si chiama infatti SWITCH in C, CASE in Pascal e non esiste in Basic (anche se vedremo come simularla). CASE non fa altro che presentare una serie di possibili opzioni e associa ad ognuna di queste una serie di istruzioni che vengono processate se e solo se la condizione è verificata.
Proviamo a fare un piccolo esempio di un CASE in Pascal:

```
CASE variabile OF
    'verde':
        writeln('Puoi Passare');
    'Arancione':
        writeln ('Attenzione, arriva il rosso');
    'Rosso':
        writeln('non si passa')
end;
```

Cosa fa il CASE appena scritto? Si comporta come un automobilista che può passare con il verde, sa di stare in guardia se esce l'arancione e si ferma nel caso ci sia rosso, pratico vero?
Riassumendo, quindi, CASE è una selezione multipla che può essere estesa a quante più condizioni si vogliono specificare.
Riprendiamo il nostro discorso di velocizzazione del software, adesso scriviamo un paio di esempi di procedure che, seppure corrette, penalizzano di molto la velocità di esecuzione:

```
SE NOME="MARIO" E COGNOME ="ROSSI"
 E ANNI=27 ECOLOREOCCHI="AZZURRO"
  ALLORA
    SCRIVI "MI RICORDO DI TE, SEI MARIO
    ROSSI"
FINE SE
```

Quanti sono i controlli effettuati?
Vi posso assicurare che sono tanti perché il computer non può fare scelte multiple, le deve in qualche modo spezzare in gruppi di 2; visto che tutte le caratteristiche devono essere contemporaneamente verificate, posso fare, ora che l'abbiamo imparato, una serie di selezioni vincolate, ossia:
(Racchiuderemo tra parentesi quadre i commenti)

```
SE NOME= "MARIO"
[abbiamo già eliminato gran parte della
popolazione italiana]
  ALLORA
    SE COGNOME="ROSSI"
     [la popolazione si riduce a qualche
     centinaio di esemplari]
      ALLORA
        SE ANNI=27
          [dovremmo già aver raggiunto il
          soggetto cercato]
          ALLORA
            SE COLOREOCCHI="AZZURRO"
              ALLORA
                SCRIVI "MI RICORDO DI TE,
                SEI MARIO ROSSI"
            FINE SE
        FINE SE
    FINE SE
FINE SE
```

Sicuramente qualcuno starà pensando che il tempo che ho impiegato per scrivere il pezzo di programma ha eliminato i vantaggi dell'algoritmo; poniamo il caso però di dover controllare tutti i cittadini italiani: certamente il guadagno è enorme!
In conclusione, quindi, valutiamo precedentemente lo spazio da campionare prima di sviluppare delle procedure che, a posteriori, potrebbero risultare troppo penalizzanti.
Quanto detto sopra, anche se gli esempi riguardavano controlli industriali e primitive ricerche su file, possiamo applicarlo al nostro vecchio "abbozzo" di videogioco; riprendiamo la struttura di controllo del movimento:

```
IF STICK(2) =  1 THEN OBJECT.X
1,OBJECT.X(1)+1
IF STICK(2) = -1 THEN OBJECT.X
1,OBJECT.X(1)-1
IF STICK(3) =  1 THEN OBJECT.Y
1,OBJECT.Y(1)+1
IF STICK(3) = -1 THEN OBJECT.Y
1,OBJECT.Y(1)-1
```

Effettivamente la situazione che si presenta è in parte differente poiché non ci sono condizioni logiche complesse, rimane comunque il fatto che la gestione è molto lenta, quindi dobbiamo studiare qualche stratagemma per porvi rimedio.
Lo studio delle condizioni logiche ci può aiutare anche in questo, notiamo innanzitutto che la macchina deve fare quattro selezioni per il movimento, anche se questo non è stato comandato.
Sarebbe meglio fare in modo che Amiga leggesse gli stati del joystick solo nel caso si verifichi una situazione particolare, cioè nel caso in cui l'utente sposti la manopola.
Notiamo che STICK() può avere due valori al suo interno, questi corrispondono al movimento orizzontale e verticale.
Analizziamo il seguente listato:

```
FOR N=1 TO 10000
  OBJECT.X 1, OBJECT.X(1) + MOVIMEN
  TOORIZZONTALE
  OBJECT.Y 1, OBJECT.Y(1) + MOVIMEN
  TOVERTICALE
  OBJECT.ON 1
   MOVIMENTOORIZZONTALE=STICK(2)
  MOVIMENTOVERTICALE=STICK(3)
NEXT N
```

Non abbiamo più selezioni, la procedura si è snellita notevolmente, il compito della macchina ora è solo quello di andare a prelevare da un indirizzo particolare (quello di controllo del joystick) il valore presente (che ovviamente la logica di controllo della manopola di comando ha impostato a 0, 1 o -1 a seconda dei casi).
Attenzione quindi in fase di sviluppo delle vostre procedure, molto spesso gli algoritmi possono essere progettati in maniera più leggibile e più veloce.
Una tecnica di programmazione, più precisamente quella strutturata, consiglia l'uso frequente di sottoprocedure che svolgono compiti ben precisi; è buona norma attenersi a queste regole, soprattutto in relazione alla leggibilità del programma finale: un conto è avere una serie chilometrica di istruzioni sequenziali, tutt'altra cosa, invece è un insieme di procedure logicamente organizzate alle quali sono "affibiati" nomi specifici riguardanti la funzione che svolgono.
AmigaBasic permette una programmazione di tipo strutturato, vi sono infatti dei comandi che permettono di definire sottoprocedure semplici o addirittura dotate di variabili locali.
Una variabile locale deve essere pensata come un "cassettino" il cui contenuto appartiene solo alla procedura in questione.
Supponiamo di voler associare, ad ogni procedura, il numero di volte che essa è stata richiamata e che all'interno vi sia presente un contatore che si incrementerà di 1 ad ogni richiamo da parte del programma principale.
Se per leggibilità del software vogliamo che tutte queste variabili abbiano lo stesso nome, basterà dichiararle come locali.
Certo, ma se il programma principale contiene una variabile con lo stesso nome?
Nessun problema, non nascono conflitti e non si verificano ambiguità, è come cercare la parola Cattedrale su un'enciclopedia e su un libro associato ad una località: sul testo generale si farà riferimento ad una non ben identificata costruzione religiosa, su una guida della città di Milano il riferimento sarà sicuramente al Duomo.
Cosa sono effettivamente le sottoprocedure?
Premesso che per semplicità non verranno prese in considerazione quelle ricorsive (che non sono comunque supportate da AmigaBasic e che hanno la prerogativa di autorichiamarsi), vediamo attraverso una serie di esempi come compiere un lavoro di normale amministrazione: dobbiamo andare a comprare il pane.
Seguendo la logica di programma-

zione più coerente, scomponiamo la nostra mansione in una serie di sottocompiti di più facile interpretazione:

```
(* Dobbiamo andare ad acquistare il pane *)
ESCI DI CASA
VAI DAL PANETTIERE
COMPRA IL PANE
TORNA A CASA.
```

Già dalla prima istruzione si nota che ESCI DI CASA è composto da una serie di azioni che possono comprendere l'aprire la porta, il chiuderla, chiamare l'ascensore, scendere e uscire dall'ingresso. Molte di queste sottoistruzioni non sono ancora "atomiche" ossia inscindibili, però abbiamo raggiunto un livello di scomposizione accettabile (scrivere "gira la chiave fino a che questa si ferma, estrarre la chiave ecc." risulta pesante e inutile).
Supposto che quello di prima sia il nostro programma, adesso posso specificare, attraverso una serie di istruzioni primitive cosa significano i costrutti come "esci di casa" e gli altri.
A questo punto non faremo altro che creare una sottoprocedura (che avrà nome proprio "esci di casa") e che conterrà delle azioni elementari o altre procedure.
In termini di AmigaBasic le sottoprocedure devono sottostare alle seguenti regole sintattiche.
Devono iniziare con la parola chiave SUB, seguita dal nome della procedura stessa, dall'elenco dei parametri formali (poi vedremo cosa sono) e dalla parola chiave STATIC.
I parametri formali non sono altri che gli argomenti che vengono passati alla sottoprocedure, per esempio radicequadrata(25) ha come parametro formale 25 (che deve essere chiuso tra parentesi).
Sui valori che vengono passati, la procedura lavora e li restituisce inalterati.
Se vogliamo però che i valori cambino, è sufficiente specificarli come SHARED nella linea seguente.

Chiarifichiamo con un esempio:

```
A=10 : B=20
CALL MEDIA (A,B,C)
PRINT "La media dei due numeri è : ";C
X=5 : Y=5 : Z=5
END

SUB MEDIA(X,Y,Z) STATIC
 Z=(X+Y)/2
END SUB
```

Dopo aver stampato il risultato della media dei due numeri il programma assegna a X,Y,Z dei valori.
Questi valori sono alterati solo nel programma principale, quelli della sottoprocedura sono rispettivamente 10, 20, 15.
Se, invece, avessimo voluto il loro valore comune con quello del programma principale, la sottoprocedura sarebbe stata così:

```
SUB MEDIA STATIC
 SHARED X,Y,Z
 Z=(X+Y)/2
END SUB
```

▲

(segue da pag.73)

Viaggio esplorativo alle... 8520

24, 39 e 40, in modo da creare le necessarie variazioni di stato logico (funzionamento) delle CIA stesse.
Chi possiede una certa dimestichezza con l'elettronica digitale, può comunque vedere gli schemi elettrici contenuti nel manuale "Introduzione all'Amiga 500 o 2000".
Un'altra considerazione da fare è relativa alla porte degli 8520: vengono utilizzate dalle periferiche (mouse, joystick, modem, tastiera, stampanti, drive) mentre sul Data Bus e Registri troviamo i segnali che dialogano con la parte "interna" di Amiga.
Visto l'uso degli 8520 nelle funzioni di interfaccia, non ci resta che tornare alle "origini" dell'articolo per rispondere a tutti coloro che hanno subito un guasto delle CIA.
Risulterà più comprensiva la domanda che sempre rivolgiamo ai nostri clienti circa l'abitudine o meno di connettere joystick, mouse, cavi stampante, interfacce MIDI o altro, anche con computer acceso.
In alcuni casi, ad esempio, le interfacce MIDI o gli adattatori per drive del C 64, richiedono ad Amiga l'alimentazione per i propri TTL, pertanto si incorre in improvvisi, seppur limitati, assorbimenti di corrente. Le CIA, occorre ricordarlo, sono integrati abbastanza complessi e non protetti, collegati direttamente alle porte esterne di Amiga (ad esclusione della RS232 che fa uso anche di due IC driver) e non "sopportano" sollecitazioni elettriche o di energia elettrostatica.
Tutte le altre conclusioni le lasciamo a voi, nella speranza di aver contribuito, senza presunzione alcuna, ad aumentare le conoscenze del vostro e nostro computer, sfatando luoghi comuni o altre teorie che, ci risulta, vengono divulgate per soli fini commerciali o di protezionismo. Se poi vi recate in un laboratorio di assistenza che ripara il vostro Amiga, provate a chiedere quale delle due CIA è stata sostituita e verificate la veridicità di quanto detto. In COMPUTER LAB, nella fase finale della riparazione, vengono testate tutte le porte, e non solo, utilizzando il Kit diagnostico già oggetto di altro articolo: l'utilità di questa operazione può essere ora meglio compresa conoscendo le molteplici funzioni svolte dagli ormai conosciutissimi 8520. ▲

I Servizi di

Amiga Magazine offre una serie di servizi per agevolare i propri lettori nel reperimento di software e materiale utile alla programmazione. E' disponibile l'intera libreria di dischetti di pubblico dominio curata da Fred Fish. Ogni dischetto contiene numerosi programmi e utility, spesso corredati da listati sorgenti e commenti degli autori.
Per districarsi fra le centinaia di programmi disponibili nei dischi di Fred Fish, come in precedenza su Transactor per Amiga, pubblicheremo nei prossimi mesi l'elenco ragionato.
Tale elenco riporta, divisi per categoria, tutti i programmi presenti, completandoli con informazioni quali la descrizione della funzione, l'autore, il numero di versione, la disponibilità del sorgente e il disco nel quale sono contenuti. I dischetti possono essere ordinati contrassegnando i numeri desiderati, purché la quantità sia di almeno 3 dischi.

BUONO D'ORDINE

Completare il buono d'ordine (o una sua fotocopia) e spedire in busta chiusa a:
I servizi di Amiga Magazine, Via Pola 9 - 20124 Milano

Si può allegare: assegno, contanti o fotocopia della ricevuta di versamento **c/c n.11666203** intestato a **Gruppo Editoriale Jackson.** Non si effettuano spedizioni in contrassegno.

Desidero ricevere i seguenti articoli; contrassegnare con una X i numeri di Fish Disk desiderati (minimo 3)

❏ 1	❏ 12	❏ 23	❏ 34	❏ 45	❏ 56	❏ 67	❏ 78	❏ 89	❏ 100	❏ 111	❏ 122	❏ 133	❏ 144	❏ 155	❏ 167
❏ 2	❏ 13	❏ 24	❏ 35	❏ 46	❏ 57	❏ 68	❏ 79	❏ 90	❏ 101	❏ 112	❏ 123	❏ 134	❏ 145	❏ 156	❏ 168
❏ 3	❏ 14	❏ 25	❏ 36	❏ 47	❏ 58	❏ 69	❏ 80	❏ 91	❏ 102	❏ 113	❏ 124	❏ 135	❏ 146	❏ 157	❏ 169
❏ 4	❏ 15	❏ 26	❏ 37	❏ 48	❏ 59	❏ 70	❏ 81	❏ 92	❏ 103	❏ 114	❏ 125	❏ 136	❏ 147	❏ 158	❏ 170
❏ 5	❏ 16	❏ 27	❏ 38	❏ 49	❏ 60	❏ 71	❏ 82	❏ 93	❏ 104	❏ 115	❏ 126	❏ 137	❏ 148	❏ 159	❏ 171
❏ 6	❏ 17	❏ 28	❏ 39	❏ 50	❏ 61	❏ 72	❏ 83	❏ 94	❏ 105	❏ 116	❏ 127	❏ 138	❏ 149	❏ 160	❏ 172
❏ 7	❏ 18	❏ 29	❏ 40	❏ 51	❏ 62	❏ 73	❏ 84	❏ 95	❏ 106	❏ 117	❏ 128	❏ 139	❏ 150	❏ 161	
❏ 8	❏ 19	❏ 30	❏ 41	❏ 52	❏ 63	❏ 74	❏ 85	❏ 96	❏ 107	❏ 118	❏ 129	❏ 140	❏ 151	❏ 162	
❏ 9	❏ 20	❏ 31	❏ 42	❏ 53	❏ 64	❏ 75	❏ 86	❏ 97	❏ 108	❏ 119	❏ 130	❏ 141	❏ 152	❏ 163	
❏ 10	❏ 21	❏ 32	❏ 43	❏ 54	❏ 65	❏ 76	❏ 87	❏ 98	❏ 109	❏ 120	❏ 131	❏ 142	❏ 153	❏ 165	
❏ 11	❏ 22	❏ 33	❏ 44	❏ 55	❏ 66	❏ 77	❏ 88	❏ 99	❏ 110	❏ 121	❏ 132	❏ 143	❏ 154	❏ 166	

Nota: il n° 164 non è disponibile

❏ 3 Fish Disk L. 24.000 (abbonati L.18.000)

❏ L. 5.000 per ogni disco aggiuntivo riferito ad ogni singolo ordine

❏ Interfaccia MIDI L. 92.000

Tutti i prezzi sono da intendersi IVA inclusa e spese di spedizione comprese.

Firma ______________________________

(se minorenne quella di un genitore) Gli ordini non firmati non verranno evasi.

Cognome ______________________________

Nome ______________________________

Via ______________________________

Cap ____________ Città ______________________________

Prov ____________ Telefono ______________________________

Codice abbonamento ______________________________

a cura di Andrea Laus

Nota dei voti

Grafica
Tiene conto della cura nel disegno dei fondali e degli sprite, dell'animazione, dello screen dei titoli.

Sonoro
Valuta la colonna sonora e gli effetti sonori.

Giocabilità
Tiene conto della varietà del gioco, delle opzioni e la possibilità di giocare in due e di eventuali scelte.

Durata
Indica per quanto tempo ci giocherete.

Generale
E' il voto generale dei valori precedenti e di tutto ciò che accompagna il gioco: confezione, manuale, gadget, ecc.

I voti vengono espressi in percentuale da 1 a 100.

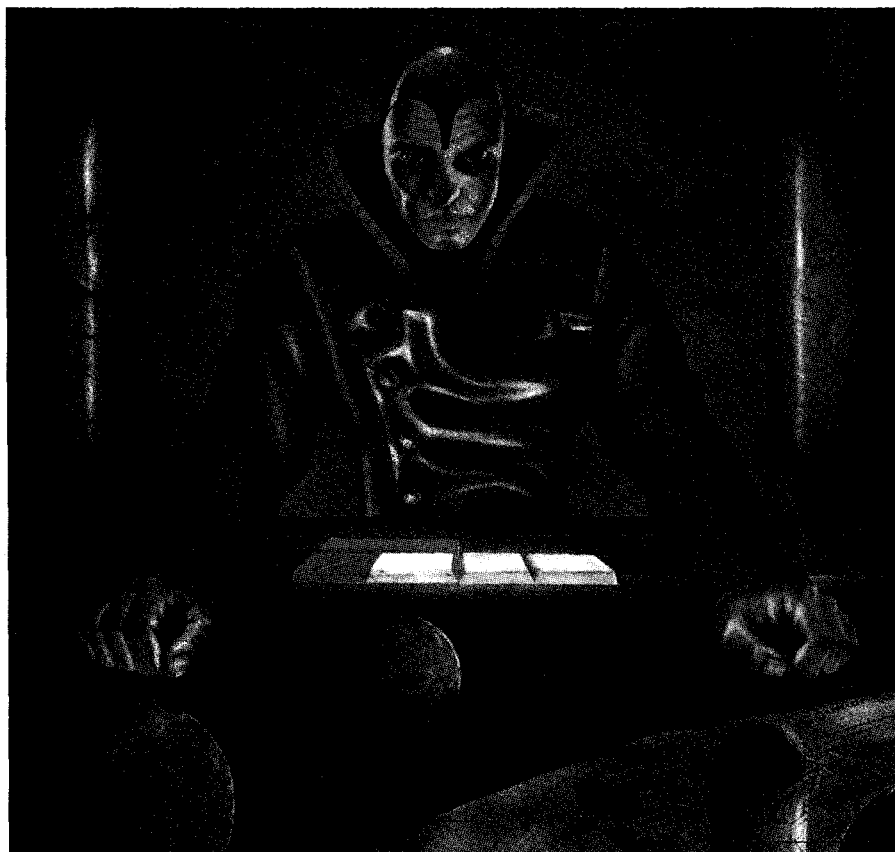
Emperor of the Mines

Grafica	65
Sonoro	70
Giocabilità	80
Durata	80
Generale	75

Software house: Impressions
Prezzo: 39.000 lire

Emperor of the Mines si presenta come un gioco in cui ciò che conta è la strategia e il fiuto per gli affari. L'ambientazione è delle più strane per un gioco di questo tipo: vi trovate nello spazio al comando della vostra flotta di astronavi.

Avete l'opportunità di dirigere una vasta organizzazione, con il semplice scopo di ottenere il maggior guadagno possibile in un determinato lasso di tempo. I profitti sono ottenuti dal denaro ricevuto per l'estrazione di minerali dalle lune presenti nel vostro sistema solare.
La durata del gioco è di circa 6 ore e, per fortuna, avete la possibilità di salvare la situazione e di riprendere il gioco in altri momenti.
Avete a disposizione tre tipi di navicelle, denominate Mark: con la prima potete atterrare sulle varie lune per fare delle trivellazioni in cerca del prezioso minerale.
Con la seconda potete trivellare in grande stile e ottenere dati sulle lune che esplorate; la terza serve per fare la manutenzione alle basi lunari. Ci sono poi veicoli trasportatori, di cui vi servirete una volta trovato l'oro, e veicoli da guerra, per difendervi dai pericolosi nemici. Grazie ai comandi disponibili sullo schermo e accessibili con il mouse, potete eseguire tutte le operazioni che vi servono, da quelle di manovra, a quelle di combattimento, da quelle di scavo a quelle di transazione commerciale. La grafica è abbastanza convincente, come lo è il sonoro; in ogni caso, la parte più interessante è la struttura stessa del gioco che non mancherà di affascinarvi.
La confezione comprende un manuale, in italiano, che, sebbene non vi spieghi come fare a diventare ricchi senza scavare, vi dà modo di conoscere a fondo il gioco.

Emperor of the Mines è distribuito da: Softel - Via E. Dal Pozzo, 7 00146 - ROMA - Tel. 06/5584334

Lost Dutchman Mine

Grafica	70
Sonoro	60
Giocabilità	60
Durata	60
Generale	65

Software house: Magnetic Images
Prezzo: 39.000 lire

Una leggenda girava nei fumosi locali dei saloon, una leggenda del vecchio West, di quelle tramandate di padre in figlio.
Racconta di una miniera, una fortuna in lingotti d'oro che aspetta solo di essere ritrovata.
Si dice che un vecchio cercatore, molto tempo prima, avesse trovato il filone madre, dove le vene d'oro sembravano illuminare tutta la terra. Purtroppo morì troppo presto e da allora nessuno è più arrivato alla miniera. Fortunatamente esiste una mappa, scritta dal vecchio cercatore, per localizzare la miniera.
Molti l'hanno cercata ma nessuno

c'è mai riuscito, ora tocca a voi ..
Vi trovate in un villaggio del vecchio west e con il joystick vi potete muovere in ogni direzione, entrare negli edifici, giocare a poker, cercare indizi. Potete anche uscire dal paese e addentrarvi nel deserto, facendo attenzione alla temperatura e ai colpi di sole.
Potete esplorare caverne, montagne e fiumi, combattendo il caldo, la fame e la sete.
Il bello di questo gioco è che non è ripetitivo, ogni nuova partita è diversa dalle precedenti. Una buona grafica e un discreto sonoro supportano la vostra avventura.
Ci sono più di cento diverse miniere e caverne da esplorare, per non dire di tutte le avventure che vi potrebbero capitare. Il manuale, incluso nella confezione, racconta la storia della vecchia miniera e vi spiega come districarvi dai pericoli.

Lost Dutchman Mine è distribuito da: Softel - Via E. Dal Pozzo, 7 00146 - ROMA - Tel. 06/5584334

Persian Gulf Inferno

Grafica	80
Sonoro	80
Giocabilità	80
Durata	80
Generale	80

Software house: Innerprise
Prezzo: 39.000 lire

Vi trovate, come dice il titolo, nel Golfo Persico, sulla più grande piattaforma petrolifera che sia mai stata costruita, alle prese con un problema non proprio da niente.
Un commando terroristico suicida ha rubato agli Stati Uniti un ordigno nucleare e si è rintanato sulla piattaforma.
Preso dalla follia, il capo del commando ha innescato il detonatore della bomba che esploderà non più tardi della mezzanotte, trasformando il Golfo in un enorme bacino radioattivo, destinato a radere al suolo tutta la civiltà occidentale.
Naturalmente chi ci va di mezzo siete sempre voi !
Dovete essere molto svelti e abili a ritrovare l'ordigno prima che scocchi l'ora fatale; tenete presente che vi rimane davvero poco tempo.
Dovete recuperare gli ostaggi, che sapranno fornirvi preziose informazioni, neutralizzare i terroristi e disinnescare l'ordigno (facile vero?)

Siete armati della vostra pistola automatica, comunque le armi e le munizioni sono sparse su tutta la piattaforma, divisa in due sezioni.
Per aprire le porte avete a disposizione una chiave davvero universale dell'esplosivo.
Il suono è la componente migliore: gli effetti digitali riproducono fedelmente le esplosioni, le voci e persino i passi.
La grafica in tre dimensioni rende perfettamente il realismo del gioco e vi regala dettagli mozzafiato.
Davvero un bel gioco, insomma, adatto soprattutto agli appassionati di missioni impossibili ai limiti della realtà.
La confezione contiene un manuale in inglese che, oltre a insegnarvi i comandi del gioco, vi racconta, in una divertente storia, l'antefatto dell'avventura che andrete a vivere

**Persian Gulf Inferno è distribuito da: Softel - Via E. Dal Pozzo, 7
00146 - ROMA
Tel. 06/5584334**

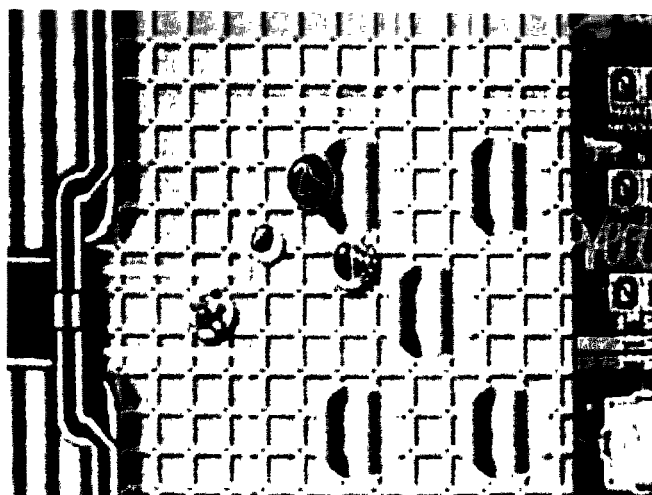
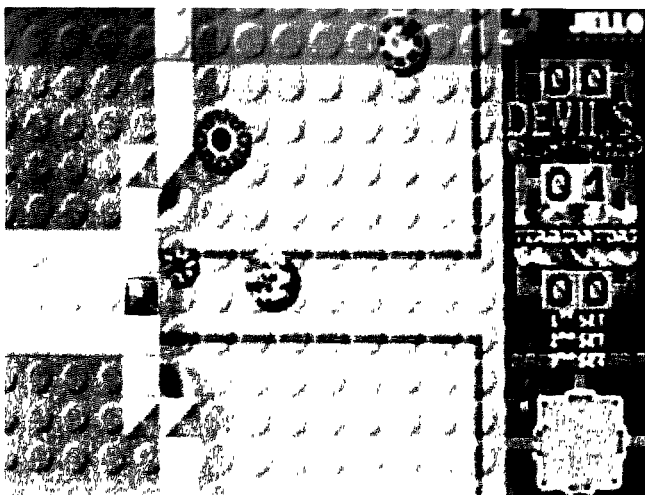
Projectyle

Grafica	68
Sonoro	80
Giocabilità	75
Durata	75
Generale	75

**Software house: Electronic Arts
Prezzo: 39.000 lire**

Projectyle è un gioco indefinibile: si ispira al calcio, nel periodo dei Mondiali è diventato di moda il video-soccer, ma non ne mantiene le regole, né tantomeno le forme.
Si tratta, come ci spiega il manuale, di un antico gioco tribale, in cui si affrontano delle squadre, meglio dire tribù, e che ha come scopo quello di mandare la palla in una delle due porte avversarie, evitando che entri in quella da voi difesa.
Da qui la somiglianza con il nostro amato sport.
Per il resto però, niente a che vedere con il calcio: il campo è formato da cinque zone quadrangolari collegate da quattro gallerie e disposte a croce; la partita inizia nel quadrato centrale e termina nelle porte, situate lungo le pareti esterne degli altri quattro quadrati.
Non ci sono falli né punizioni: l'unica

cosa che conta è mandare la palla nella porta (avversaria, ovviamente). Quando segnate un goal, il computer vi mostra l'azione al rallentatore e se doveste fare un autogoal, il punto verrebbe assegnato all'ultimo giocatore che ha toccato la palla prima di voi.
Anche se, in effetti, assomiglia poco al calcio tradizionale, ne mantiene lo spirito e questo lo rende davvero eccitante. L'azione è rapida e sciolta e non riuscirete a stancarvi facilmente di giocare.
Il punto di forza del gioco è la musica: si può scegliere fra molti motivetti quello che più ci piace (e sono tutti molto belli).
La grafica è invece meno curata.
La confezione comprende un poster-manuale che vi spiega tutto quello che c'è da sapere su questo emozionante gioco.

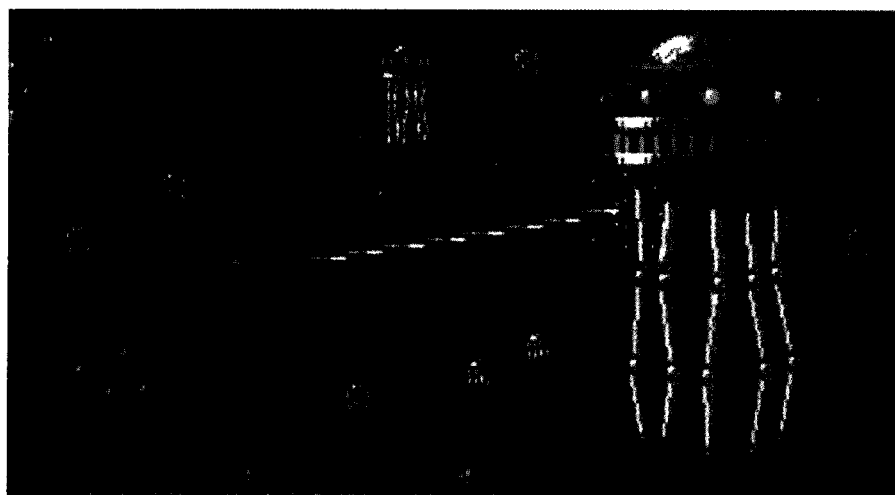
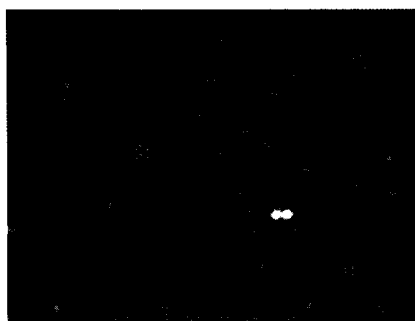
**Projectyle è distribuito da:
C.T.O. s.r.l. - Via Piemonte, 7/F
40069 Zola Predosa (BO)
Tel. 051/753133**

Datastorm

Grafica	60
Sonoro	60
Giocabilità	65
Durata	55
Generale	60

Software house: Visionary Design Technologies Inc.
Prezzo: 39.000 lire

Ecco finalmente un classico gioco Shoot'em-up, dove ciò che conta è la velocità della vostra azione. Lo scopo di Datastorm, è quello di proteggere i vostri baccelli durante il loro pericoloso viaggio sulla superficie di uno strano pianeta. Per fare questo, dovete raccoglierli uno per uno e trasportarli fino al portone. Alla fine di ogni livello, vi viene assegnato un bonus in funzione di quanti baccelli avete salvato e, ogni volta, ve ne vengono affidati in più.
Vi chiederete, dov'è la difficoltà?
Il fatto è che in questo strano pianeta non siete soli: è abitato da uno strano popolo, i Landers, avido di baccelli, che farà il possibile per battervi sul tempo nella raccolta.
Quando un Lander riesce a raggiungere la stratosfera con un baccello, si trasforma in un avversario ancora

più terribile. Inoltre, se vengono a mancare i baccelli, i nemici se la prenderanno con voi, diventando più veloci e più cattivi e... indovinate chi sarà a farne le spese?
La confezione comprende il dischetto e un breve manuale in cui vengono spiegati i vari comandi.
Si gioca con il Joystick e, visto il tipo di gioco, l'azione è molto veloce, quindi occhio ai riflessi!
La grafica e il sonoro non sono molto stimolanti, ma nella confusione, forse, non ci farete neppure caso.

Datastorm è distribuito da: Softel

All Dogs go to Heaven

Grafica	75
Sonoro	70
Giocabilità	65
Durata	65
Generale	65

Software house: Polarware
Prezzo: 49.000 lire

Questo gioco è la trasposizione elettronica del famoso film che tutti i bambini, ma non solo loro, ricorderanno: Tutti i cani vanno in paradiso. Il gioco si snoda attraverso dieci differenti schermi, dieci diverse avventure per rivivere da zero il film che vi ha fatto sognare.
Il programma, ad una prima occhiata, sembra non deludere le aspettative anche degli utenti più esigenti: c'è la Rat-Race, la corsa ad ostacoli; il Watch Arcade, un gioco per chi ha nervi d'acciaio; Catch The Flea, per la coordinazione; la Dog Fight, una battaglia ninja fra cani, e altre avventure altrettanto altisonanti. In effetti, dopo averlo provato, questo gioco si presenta come un ottimo cartone animato, ma non come un gioco molto appassionante, a causa della lentezza dell'azione, soprattutto in alcune delle dieci avventure.
In ogni caso non mancano effetti sonori e grafici straordinari, degni del migliore Don Bluth, il disegnatore di Dragon's Lair che, ancora una volta ci mostra il suo talento. Il manuale offre una dettagliata descrizione delle dieci avventure.

All Dogs Go to Heaven è distribuito da: Leader Distribuzione
Via Mazzini, 15
Casciago (Va)
Tel.0332/212255